Computationally Feasible Approaches to Automated Mechanism Design

by

Mingyu Guo

Department of Computer Science
Duke University

Date: ____________________

Approved:

____
Vincent Conitzer, Advisor

____
Kamesh Munagala

____
David Parkes

____
Ronald Parr

Dissertation submitted in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in the Department of Computer Science
in the Graduate School of Duke University
2010

Abstract

(Computer Science)

Computationally Feasible Approaches to Automated Mechanism Design

by

Mingyu Guo

Department of Computer Science
Duke University

Date: ____________________

Approved:

____
Vincent Conitzer, Advisor

____
Kamesh Munagala

____
David Parkes

____
Ronald Parr

An abstract of a dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Computer Science in the Graduate School of Duke University
2010

Abstract

In many multiagent settings, a decision must be made based on the preferences of multiple agents, and agents may lie about their preferences if this is to their benefit. In mechanism design, the goal is to design procedures (mechanisms) for making the decision that work in spite of such strategic behavior, usually by making untruthful behavior suboptimal. In automated mechanism design, the idea is to computationally search through the space of feasible mechanisms, rather than to design them analytically by hand. Unfortunately, the most straightforward approach to automated mechanism design does not scale to large instances, because it requires searching over a very large space of possible functions. In this dissertation, we adopt an approach to automated mechanism design that is computationally feasible. Instead of optimizing over all feasible mechanisms, we carefully choose a parameterized subfamily of mechanisms. Then we optimize over mechanisms within this family. Finally, we analyze whether and to what extent the resulting mechanism is suboptimal outside the subfamily. We apply (computationally feasible) automated mechanism design to three resource allocation mechanism design problems: mechanisms that redistribute revenue, mechanisms that involve no payments at all, and mechanisms that guard against false-name manipulation.

Contents

List of Tables

List of Figures

List of Abbreviations and Symbols

Symbols

$I = \{1, 2, \ldots, n\}$	The set of agents.
$-i$	The set of agents other than agent i
$G = \{1, 2, \ldots, m\}$	The set of items.
O	The outcome space.
Θ	The type space.
$\theta_i \in \Theta$	Agent i's reported type. (Since we consider only truthful mechanisms, when there is no ambiguity, we also use θ_i to denote i's true type.)
$\theta_{-i} \in \Theta^{n-1}$	The types reported by agents other than agent i.

Abbreviations

VCG	Vickrey-Clarke-Groves mechanism
AMD	Automated mechanism design
CFAMD	Computationally feasible automated mechanism design
WCO	Worst-case optimal VCG redistribution mechanism
OEL	Optimal-in-expectation linear VCG redistribution mechanism
SP	Strategy-proof mechanisms
SD	Swap-dictatorial mechanisms
IP	Increasing-price mechanisms
LIP	Linear increasing-price mechanisms

FNP	False-name-proofness
FNPW	False-name-proofness with bid withdrawal
PORF	Price-oriented, rationing-free
OBDNH	Others' bids do not help
NSA	No superadditive price increase
PIA	Prices increase with agents
NSAW	No superadditive price increase with withdrawal
S-NSAW	Sufficient condition for no superadditive price increase with withdrawal
MMVIP	Maximum marginal value item pricing mechanism
IIG	Independence of irrelevant goods

Acknowledgements

My foremost gratitude goes to my thesis advisor Prof. Vincent Conitzer. I would like to thank Vince for introducing me to the field of computational game theory and mechanism design, for always being available, supportive, and encouraging, for all the great advice he shared with me, for his invaluable support in my job search, and most importantly, for shaping me as a researcher. I am very lucky to be Vince's first Ph.D. advisee in computer science, and had the luxury to enjoy extra support in every stage of my Ph.D. study. Besides being my academic advisor, Vince has also been a great friend who influenced me positively in other aspects of life.

I also owe my gratitude to my other committee members for their advice and support. Besides thanking their roles as my committee members, I would like to thank Prof. Kamesh Munagala for initiating the weekly CS-Econ reading group, which is a great forum for exchanging ideas; I would like to thank Prof. David Parkes at Harvard for continuously encouraging and advising me since the early stage of my Ph.D. study, and for his solid support in my job search; I would like to thank Prof. Ronald Parr for introducing me to AI, and helping me make my research more accessible to people in general AI. I also would like to thank Prof. Giuseppe Lopomo for serving as the committee member of my Research Initiation Project.

I would like to thank Dr. David Pennock at Yahoo! Research for his guidance during my internship there. I benefited a lot from Dave's expertise on prediction markets. I am also grateful for Dave's support in my job search. I would like to thank Prof. Hervé Moulin at Rice University for sharing with me his perspective as an economist, and his support in my job search.

I am also grateful for all my colleagues and friends at Duke. I am grateful especially to the participants of the CS-Econ reading group, the DRIV seminar, and the micro theory lunch seminar, for their valuable feedback on my research, and especially to the recently retired graduate program coordinator Diane Riggs, for keeping an eye on my progress.

I owe a lot to my parents. As a second child, I cost them a significant amount of fortune due to the one-child policy of China. I would like to thank my mother for always standing behind me and keeping my childhood home for me. I would like to thank my father for providing me my first computer when computers were relatively rare in China, and driving me to the first Internet cafe in my home city. I also would like to thank my elderly brother for taking care of my parents while I am away.

Finally, I thank my girlfriend Beibei who stood beside me through the most difficult period of my Ph.D. journey. She taught me the true meaning of love. Without her many sacrifices I couldn't be here. She helped me through so many milestones and deadlines: WINE, AAMAS, job application, EC, AAAI, prelim, job interview, defense. Beibei took extremely good care of me in the past year (with the help of her mother, to whom I am also deeply indebted to). Getting to know Beibei is the best thing that Duke brought me.

1

Introduction

In the past decade, computer scientists have been increasingly focusing on interdisciplinary topics that lie in the intersection of computer science and economics, in large part because the development of the Internet has led to novel and influential electronic markets, accompanied by new computing challenges. For example, *keyword auctions* (Internet advertisement auctions), such as Google AdWords and Yahoo! Sponsored Search, form a multi-billion dollar industry in rapid growth. Their unique features, including their computational aspects, distinguish them from traditional auctions, thus calling for new auction protocols [74]. On the other hand, existing markets have also benefited from the development of computing. For example, *combinatorial auctions* [40] allow market participants to express richer preferences (*e.g.*, "I want either product A or product B, but not both") than traditional auctions, which greatly increases economic efficiency when matching buyers and sellers. Besides applications on Wall Street (*e.g.*, conditional contracts), combinatorial auctions have also been used for spectrum auctions, airport takeoff and landing slot auctions, supply chain auctions, and many others. These combinatorial auctions, as well as many other novel market protocols, are made possible by modern computers and algorithms.

This dissertation studies the problem of designing new protocols, often called *mechanisms*, that implement desirable social decisions in systems involving multiple agents. Our main focus is on designing new mechanisms, specifically, new resource allocation mechanisms, with the help of computational techniques. We also address mechanism design problems that arise specifically in computer science domains.

1.1 Mechanism Design Preliminaries

Mechanism design deals with making social decisions in systems involving multiple agents. A typical setting in mechanism design is given by the following. There is a set of *agents* $I = \{1, 2, \ldots, n\}$, and a set of possible *outcomes* O (social decisions). For example, in resource allocation problems, an outcome specifies who wins which resources. We generally assume that the agents are rational in a game-theoretic sense, and that each agent's preferences are private information for that agent. Let Θ be the space of all possible *types* that agents may have, where agent i's type θ_i contains all of agent i's private information. For example, in a single-item auction, agent i's type θ_i is a nonnegative real value, which is i's valuation for winning the only item. In a combinatorial auction [40] in which a set of items S are simultaneously for sale, in general, θ_i consists of $2^{|S|} - 1$ nonnegative real numbers, where each number represents the valuation for receiving a certain nonempty bundle (subset) of the items. Often, the type space is assumed to be more restricted. For example, if each agent is only interested in a single bundle (that is, agents are *single-minded*), then a type θ consists of a pair (S', x), where S' is the bundle that the agent is interested in, and her valuation equals x if the bundle she wins contains S' (and her valuation is 0 otherwise). Another special case is a *multi-unit* auction, in which m indistinguishable items are for sale (equivalently, there are multiple units of the same item for sale). Here, a type consists of m nonnegative real numbers, where the

jth number indicates the value for obtaining j units. A special case is a multi-unit auction with *unit demand*, in which each agent wants to obtain only one unit—that is, all m numbers are always the same, so a type effectively consists of a single number. As is common, we assume that preferences are *quasi-linear*, that is, agent i's utility equals to her valuation for the items that she wins, minus her payment (in settings where payments are allowed).

We generally focus on *direct-revelation* mechanisms, in which each agent makes a report $\theta_i \in \Theta$ of her preferences to the mechanism, which then makes the decision based on these reported types. Hence, a mechanism is a function $f : \Theta^n \to O$. (If randomized mechanisms are allowed, then a mechanism is a function $f : \Theta^n \to \Delta(O)$, where $\Delta(O)$ are the probability distributions over O.) In settings where payments are allowed, a mechanism also needs to specify how much each agent pays. That is, in these settings, a mechanism is determined by both an allocation function f and a payment function p ($p = (p_1, p_2, \ldots, p_n) : \Theta^n \to \mathbb{R}^n$). When an agent reports, she may lie so that her reported type may not be exactly θ_i. By a result known as the *revelation principle* [49, 51, 85, 86], we can restrict attention to direct-revelation mechanisms that incentivize truthful reporting.

Let $u_i(\theta_i, o)$ be the valuation that agent i obtains if she has true type θ_i and the outcome is o. According to the quasi-linear assumption, agent i's utility equals her valuation minus her payment.

A mechanism is *strategy-proof* if each agent is best off reporting her type truthfully, no matter what her type is and no matter what the other agents report. That is, for all $(\theta_1, \ldots, \theta_n) \in \Theta^n$ and all $\hat{\theta}_i \in \Theta$,

$$u_i(\theta_i, f(\theta_1, \ldots, \theta_i, \ldots, \theta_n)) - p_i(\theta_i, \ldots, \theta_i, \ldots, \theta_n) \geq$$

$$u_i(\theta_i, f(\theta_1, \ldots, \hat{\theta}_i, \ldots, \theta_n)) - p_i(\theta_i, \ldots, \hat{\theta}_i, \ldots, \theta_n).$$

A mechanism is *efficient* if its outcome always maximizes the agents' total valuation. That is, for all $(\theta_1, \ldots, \theta_n) \in \Theta^n$,

$$f(\theta_1, \ldots, \theta_n) \in \arg\max_o \{\sum_i u_i(\theta_i, o)\}.$$

A mechanism is *(ex post) individually rational (IR)* if each agent's utility is always nonnegative (as long as she reports truthfully), so that participating is always optimal. That is, for all $(\theta_1, \ldots, \theta_n) \in \Theta^n$, for all i,

$$u_i(\theta_i, f(\theta_1, \ldots, \theta_i, \ldots, \theta_n)) - p_i(\theta_1, \ldots, \theta_i, \ldots, \theta_n) \geq 0.$$

A mechanism is *(strongly) budget balanced* if the agents' total payment is always 0. That is, for all $(\theta_1, \ldots, \theta_n) \in \Theta^n$,

$$\sum_i p_i(\theta_1, \ldots, \theta_i, \ldots, \theta_n) = 0.$$

A mechanism is *non-deficit* if the agents' total payment is always at least 0. That is, for all $(\theta_1, \ldots, \theta_n) \in \Theta^n$,

$$\sum_i p_i(\theta_1, \ldots, \theta_i, \ldots, \theta_n) \geq 0.$$

Perhaps the most famous mechanism is the *Vickrey-Clarke-Groves (VCG)* mechanism [103, 25, 52]. This mechanism chooses an outcome o^* that maximizes the agents' total valuation, that is, $o^* \in \arg\max_o \{\sum_i u_i(\theta_i, o)\}$. That is, the mechanism is *efficient*. Then, to determine agent i's payment, it computes an outcome o^*_{-i} that would have been optimal if agent i had not been present, that is, $o^*_{-i} \in \arg\max_o \{\sum_{j \neq i} u_j(\theta_j, o)\}$. Finally, it determines agent i's payment as $\sum_{j \neq i} u_j(\theta_j, o^*_{-i}) - \sum_{j \neq i} u_j(\theta_j, o^*)$. That is, agent i pays how much she "hurts" the other agents by her presence. This mechanism is well-known to be strategy-proof and efficient. Under certain minimal assumptions, it is also individually rational and non-deficit.

The goal of mechanism design is to design a mechanism that satisfies a set of desired properties, and performs well according to some objective. Example objectives include maximizing expected/worst-case *social welfare* (sum of the agents' utilities) as well as maximizing expected/worst-case *revenue* (total payment collected from the agents by the mechanism).

1.2 Automated Mechanism Design

Traditionally, economists have designed mechanisms manually, based on analytical techniques. This has resulted in many good mechanisms, such as the VCG mechanism. However, there are many different variants of the mechanism design problem, depending on the specific desired properties and objective, and many of these variants remain unsolved. Often, they correspond to very difficult optimization problems, and the solutions can take very complex forms.

This is where automated mechanism design can be of help. The first precise general formulation of the automated mechanism design problem was given by Conitzer and Sandholm [30]. The basic idea is to solve for the function f as a constrained optimization problem, where the desired properties (e.g., strategy-proofness, IR) correspond to the constraints in the optimization, and the objective (e.g., social welfare, revenue) corresponds to the objective in the optimization.

To illustrate the framework, suppose that we have a prior distribution p for the true type vector of the agents, $p : \Theta^n \to [0, 1]$. For simplicity, let us for now assume that we are dealing with a setting where payments are not allowed. That is, $u_i(\theta, o)$ is the utility that agent i obtains if she has true type θ and the outcome is o. Finally, suppose (for the sake of example) that strategy-proofness is the only desired property, we want the mechanism to be deterministic, and we aim to maximize expected social welfare. We obtain the following general formulation:

Variable function: $f : \Theta^n \to O$

Maximize $\sum_{\vec{\theta} \in \Theta^n} p(\vec{\theta}) \sum_{i=1}^{n} u_i(\theta_i, f(\vec{\theta}))$ (expected social welfare)

Subject to: $\forall i \in \{1, 2, \ldots, n\}$, $\vec{\theta} \in \Theta^n, \theta_i' \in \Theta$,
$u_i(\theta_i, f(\vec{\theta})) \geq u_i(\theta_i, f(\theta_1, \ldots, \theta_{i-1}, \theta_i', \theta_{i+1}, \ldots, \theta_n))$ (strategy-proofness)

A key problem is that this is a problem of optimizing a *function*. Perhaps the most basic approach to solving such problems is the one explored in a sequence of papers by Conitzer and Sandholm [30, 32, 31, 34] (an overview can be found in a chapter by Conitzer [27]); it works as follows. If the type space Θ and the outcome space O are finite, and we allow for randomized mechanisms, then it is possible to write the optimization problem as a linear program: this is done by defining a probability variable $p_f(o|\vec{\theta})$ for each $\vec{\theta} \in \Theta^n$ and $o \in O$. For the case where we require a deterministic mechanism, we can add the constraint that each of these probabilities must be in $\{0, 1\}$ to obtain an integer program (in the deterministic case, the problem is generally NP-hard even with one agent). Still, the scalability of this approach is very limited. One reason is that both the number of variables and the number of constraints are exponential in n. Another problem is that type spaces are generally not finite and require discretization for this approach to work. For small instances, this approach is feasible, and the solutions to these small instances will sometimes allow us to conjecture more general results, but generally the limitations on scalability are too constraining.

1.3 Previous Research on Automated Mechanism Design

Since proposed, the principle of automated mechanism design (broadly interpreted) has been applied to various settings. Constantin and Parkes [37] applied automated mechanism design to the problem of designing revenue-optimal dynamic single-item auctions with interdependent-value agents. Their (mixed-integer programming) formulation turned out to be computationally feasible under several assumptions, including that the number of agents is small, and the type space is small (coarse discretization). Jurca and Faltings [71] applied automated mechanism design to compute the minimum payments that make a reputation mechanism incentive-compatible. In their model, when designing such mechanisms, we only need to consider two agents at a time. That is, in this particular problem, as long as the type space is small, the automated mechanism design process is computationally feasible. Bhattacharya *et al.* [12] applied automated mechanism design to solve for revenue-maximizing multi-item auctions with budget constrained agents. They relied on linear program relaxations to derive upper bounds on performance, and used rounding schemes to construct feasible mechanisms whose performances approximate the obtained upper bounds. Hyafil and Boutilier [68] studied mechanisms that only require partial revelation of preferences from agents. They discussed how to use direct AMD to design such mechanisms. Sandholm *et al.* [98] applied automated mechanism design to the design of multistage mechanisms. Here, they also do not require complete revelation of preferences from agents. Instead, agents are queried sequentially, and only about information that is relevant given previous query responses. Automated mechanism design has also been applied to the online mechanism design problem [62] and to analyze existing mechanism design theorems (Arrow's Impossibility Theorem [79] and the Myerson-Satterthwaite Impossibility Theorem [88]).

Automated mechanism design has also been used for optimizing over specific families of parameterized mechanisms. For example, Likhodedov and Sandholm [78, 77] studied the problem of constructing revenue-maximizing combinatorial auctions. They focused their attention on the family of *Affine Maximizer Auctions* and the family of *Virtual Valuations Combinatorial Auctions*. Mechanisms inside these two families are characterized by a set of parameters. They proposed several algorithms for searching for the optimal values for the parameters. Likhodedov and Sandholm [76] studied the problem of designing efficiency-maximizing multi-unit auctions, subject to a minimal revenue constraint. They showed that the optimal mechanism belongs to a family of mechanisms parameterized by a single parameter, and gave a binary search algorithm for identifying the optimal parameter. Sandholm and Gilpin [99] studied a special kind of mechanisms that consist of sequences of take-it-or-leave-it offers. These mechanisms form a mechanism family parameterized by the "offers". They evaluated mechanisms based on equilibrium analysis, and proposed an algorithm for optimizing the parameters (offers). Vorobeychik *et al.* [105] focused on the family of *Shared-Good Auctions*. Each mechanism within the family is characterized by two parameters. For different objectives, they searched for the optimal shared-good auctions within the family. There is also a series of papers on simultaneously optimizing for parameters of the non-strategy-proof mechanisms and the agents' equilibrium strategies [26, 93, 18, 104, 80]. Our approach in this dissertation is also based on the idea of optimizing over families of parameterized mechanisms.

Another line of research called *heuristic mechanism design* also deals with designing mechanisms with the help of computation [92, 38, 91]. Unlike in automated mechanism design where the problem of designing mechanisms is treated as an optimization problem, in heuristic mechanism design, we start from a heuristic mechanism (a mechanism that we expect to perform reasonably well), then we rely on computation to modify the heuristic mechanism so that it becomes truthful – a process that is called *self-correction/output ironing*. An approach that is similar to heuristic mechanism design is *incremental mechanism design*, which was proposed by Conitzer and Sandholm [36]. In incremental mechanism design, we start with a naïve mechanism that is manipulable, and then incrementally make it more strategy-proof over iterations.

1.4 Computationally Feasible Automated Mechanism Design

To address the scalability issue of automated mechanism design, we adopt the following approach in this dissertation. We call this approach *computationally feasible automated mechanism design (CFAMD)*.

- For a specific setting, let Ω denote the set of all feasible mechanisms.[1]
- Instead of optimizing over Ω directly, we first choose a suitable subfamily $\Omega' \subseteq \Omega$ based on analytical considerations, where the mechanisms in Ω' can be parameterized using k parameters, so that a vector $(c_1, \ldots, c_k)$ defines a mechanism $f_{c_1,c_2,\ldots,c_k} \in \Omega'$.
- We then optimize over $(c_1, \ldots, c_k)$, which is generally a much easier problem.[2]
- Finally, we analytically study the suboptimality of the resulting mechanism in the full set Ω.

Unlike in the more basic automated mechanism design approach described earlier, this approach requires significant human input: we need to find a good subfamily Ω' as well as a formulation/algorithm for optimizing over this subfamily, and in the end we need to analyze the suboptimality of the resulting mechanism by hand. From the perspective of artificial intelligence, it may be disappointing that the approach requires so much domain-specific expertise from humans. On the other hand, we have been much more successful at contributing new results to microeconomic theory

[1] Throughout this dissertation, we only consider truthful mechanisms. That is, feasibility includes truthfulness.

[2] As discussed in the previous section, there have been some previous automated mechanism design papers that optimize over a parameterized family (notably, [78, 77, 76, 99, 105]), and that also have other aspects in common with the CFAMD approach layed out here—for example, some of these papers give some analytical justification for the class of mechanisms that they consider. Therefore, we certainly do not claim to be the first to use some of these ideas. However, to our knowledge, we are the first to lay out this general CFAMD approach in the abstract.

with this human-machine interactive approach, so we believe that at this point, this is the best way in which artificial intelligence can make contributions to the theory of mechanism design.

The key step is choosing Ω'. If Ω' is too restrictive, then even if we find the optimal mechanism in it, its performance will be too suboptimal. If Ω' is too general, then the optimization problem becomes too difficult again. That is, we want Ω' to be general enough that an (almost) optimal mechanism exists in Ω', and Ω' to be specific enough that we know how to optimize over its parameters. Of course, choosing a good Ω' is not as simple as doing a binary search on the specificity-generality spectrum: an unfortunate choice may result in an Ω' that is difficult to optimize over and still does not produce good mechanisms![3] The point is that there is some "art" involved in choosing a good Ω'.

1.5 Contribution and Organization

My dissertation statement is that the CFAMD approach can be successfully employed to discover new results in the theory of mechanism design.

In Section 1.4, we formalized the CFAMD approach. This section is based on [57].

In Chapter 2, we apply CFAMD to the problem of designing resource allocation mechanisms that redistribute their revenue back to the agents. For allocation problems with one or more items, the well-known Vickrey-Clarke-Groves (VCG) mechanism (aka. Clarke mechanism, Generalized Vickrey Auction) as introduced in Section 1.1 is efficient, strategy-proof, individually rational, and does not incur a deficit. However, it is not (strongly) budget balanced: generally, the agents' payments will sum to more than 0. In this chapter, we study mechanisms that redistribute some of

[3] It should be noted that our approach is subject to the *analytical bottleneck* as specified in [91]. That is, sometimes, we may not be able to find a good Ω'.

the VCG payments back to the agents, while maintaining the desirable properties of the VCG mechanism. Below is a detailed description of this chapter's contribution and organization:

- Section 2.1: We study the problem of designing VCG redistribution mechanisms that redistribute the most in the *worst case*. For auctions with multiple indistinguishable units in which marginal values are nonincreasing, we derive a mechanism that is optimal in this sense. We also derive an optimal mechanism for the case where we drop the non-deficit requirement. Finally, we show that if marginal values are not required to be nonincreasing, then the original VCG mechanism is worst-case optimal. This section is based on [56].

- Section 2.2: We study the problem of designing VCG redistribution mechanisms that redistribute the most in *expectation* when prior distributions over the agents' valuations are available. For auctions with multiple indistinguishable units in which each agent is only interested in one unit, we analytically derive a mechanism that is optimal among linear redistribution mechanisms. We also propose discretized redistribution mechanisms. We show how to automatically solve for the optimal discretized redistribution mechanism for a given discretization step size, and show that the resulting mechanisms converge to optimality as the step size goes to zero. We present experimental results showing that for auctions with many bidders, the optimal linear redistribution mechanism redistributes almost everything, whereas for auctions with few bidders, we can solve for the optimal discretized redistribution mechanism with a very small step size. We then generalize our setting to auctions with multiple indistinguishable units in which marginal values are nonincreasing. We extend the notion of linear redistribution mechanisms to this more general setting. We introduce a linear program for finding the optimal linear redistribution mechanism. Since this linear program is unwieldy, we also introduce one simplified linear program that produces relatively good linear redistribution mechanisms. Finally, we conjecture an analytical solution for the simplified linear program. This section is based on [59].

- Section 2.3: We study the problem of designing mechanisms whose redistribution functions are *undominated* in the sense that no other mechanisms can always perform as well, and sometimes better. (Here, "always" means for every profile of types.) We introduce two measures for comparing two VCG redistribution mechanisms with respect to how well off they make the agents. We say a non-deficit VCG redistribution mechanism is individually undominated if there exists no other non-deficit VCG redistribution mechanism that always has a larger or equal redistribution for each agent. We say a non-deficit VCG redistribution mechanism is collectively undominated if there exists no other non-deficit VCG redistribution mechanism that always has a larger or equal sum of redistributions. We study the question of finding maximal elements in the space of non-deficit redistribution mechanisms, with respect to the partial orders induced by both measures. For the first measure, we give a characterization of all individually undominated VCG redistribution mechanisms, and propose two techniques for generating individually undominated mechanisms based on known individually dominated mechanisms. Experimental results show that these techniques can significantly increase the agents' utilities. For the second measure, we characterize all collectively undominated VCG redistribution mechanisms that are anonymous and have linear payment functions, for auctions with multiple indistinguishable units, where each agent is only interested in a single copy of the unit. This section is based on [55, 3].[4]

[4] [3] is joint work with Krzysztof Apt and Evangelos Markakis. This dissertation does not contain materialsthat were contributed by Apt or Markakis.

- Section 2.4: We study the problem of designing the allocation rule together with the redistribution scheme, allowing for the allocation to be inefficient. We notice that sometimes even the optimal VCG redistribution mechanism results in a low total utility for the agents, even though the items are allocated efficiently. We further notice that by allocating inefficiently, more payment can sometimes be redistributed, so that the net effect is an increase in the sum of the agents' utilities. Our objective is to design mechanisms that are competitive with the first-best allocation. We define linear allocation mechanisms. We propose an optimization model for simultaneously finding an allocation mechanism and a payment redistribution rule which together are optimal, given that the allocation mechanism is required to be either one of, or a mixture of, a finite set of specified linear allocation mechanisms. Finally, we propose several specific (linear) mechanisms that are based on burning items, excluding agents, and (most generally) partitioning the items and agents into groups. We show or conjecture that these mechanisms are optimal among various classes of mechanisms. This section is based on [54].

In Chapter 3, we apply CFAMD to the problem of designing resource allocation mechanisms that do not rely on payments at all. This is useful in settings where no currency has (yet) been established (as may be the case, for example, in a peer-to-peer network, as well as in many other multiagent systems); or where payments are prohibited by law; or where payments are otherwise inconvenient. Below is a detailed description of this chapter's contribution and organization:

- Section 3.1: We study the problem of allocating a single item repeatedly among multiple competing agents, in an environment where monetary transfers are not possible. We design (Bayes-Nash) incentive compatible mechanisms that do not rely on payments, with the goal of maximizing expected social welfare.

We first focus on the case of two agents. We introduce an artificial payment system, which enables us to construct repeated allocation mechanisms without payments based on one-shot allocation mechanisms with payments. Under certain restrictions on the discount factor, we propose several repeated allocation mechanisms based on artificial payments. For the simple model in which the agents' valuations are either high or low, the mechanism we propose is 0.94-competitive against the optimal allocation mechanism with payments. For the general case of any prior distribution, the mechanism we propose is 0.85-competitive. We generalize the mechanism to cases of three or more agents. For any number of agents, the mechanism we obtain is at least 0.75-competitive. The obtained competitive ratios imply that for repeated allocation, artificial payments may be used to replace real monetary payments, without incurring too much loss in social welfare. This section is based on [61].[5]

- Section 3.2: We investigate the problem of allocating items among competing agents in a (single-round) setting that is both prior-free and payment-free. Specifically, we focus on allocating multiple heterogeneous items between two agents with additive valuation functions. Our objective is to design strategy-proof mechanisms that are competitive against the most efficient (first-best) allocation. We introduce the family of linear increasing-price (LIP) mechanisms. The LIP mechanisms are strategy-proof, prior-free, and payment-free, and they are exactly the increasing-price mechanisms satisfying a strong responsiveness property. We show how to solve for competitive mechanisms within the LIP family. For the case of two items, we find a LIP mechanism whose competitive ratio is near optimal (the achieved competitive ratio is 0.828, while any strategy-proof mechanism is at most 0.841-competitive). As the number of

[5] [61] is joint work with Daniel Reeves. This dissertation does not contain materials that were contributed by Reeves.

items goes to infinity, we prove a negative result that any increasing-price mechanism (linear or nonlinear) has a maximal competitive ratio of 0.5. Our results imply that in some cases, it is possible to design good allocation mechanisms without payments and without priors. This section is based on [60].

In Chapter 4, we study the following manipulation in Internet auctions: an agent can submit multiple false-name bids, but then, once the allocation and payments have been decided, withdraw some of her false-name identities (have some of her false-name identities refuse to pay). While these withdrawn identities will not obtain the items they won, their initial presence may have been beneficial to the agent's other identities. We define a mechanism to be *false-name-proof with withdrawal (FNPW)* if the aforementioned manipulation is never beneficial. We first give a necessary and sufficient condition on the type space for the VCG mechanism to be FNPW. We then characterize both the payment rules and the allocation rules of FNPW mechanisms in general combinatorial auctions. Based on the characterization of the payment rules, we derive a condition that is sufficient for a mechanism to be FNPW. We also propose the *maximum marginal value item pricing (MMVIP)* mechanism. We show that MMVIP is FNPW and exhibit some of its desirable properties. We then propose an automated mechanism design technique that transforms any feasible mechanism into an FNPW mechanism, and prove some basic properties about this technique. Toward the end, we prove a strict upper bound on the worst-case efficiency ratio of FNPW mechanisms. We conclude with a characterization of FNP(W) social choice rules. This chapter is based on [58].

Finally, in Chapter 5, we discuss future research and conclude.

2

Redistribution Mechanisms

Many important problems in multiagent systems can be seen as resource allocation problems. One natural way of allocating resources among agents is to auction off the items. An allocation mechanism (or *auction*) takes as input the agents' reported valuations for the items, and as output produces an allocation of the items to the agents, as well as payments to be made by or to the agents. As was defined in Section 1.1, a mechanism is *strategy-proof* if it is a dominant strategy for the agents to report their true valuations—that is, regardless of what the other agents do, an agent is best off reporting her true valuation. A mechanism is *efficient* if it always chooses an allocation that maximizes the sum of the agents' valuations (aka. the *social welfare*).

The well-known *VCG (Vickrey-Clarke-Groves)* mechanism [103, 25, 52] is both strategy-proof and efficient.[1] In fact, in sufficiently general settings, the wider but closely related class of Groves mechanisms coincides exactly with the class of mech-

[1] We use the term "VCG mechanism" to refer to the Clarke mechanism. Sometimes people refer to the wider class of Groves mechanisms as "VCG mechanisms," but we will avoid this usage in this chapter. In fact, the mechanisms proposed in this chapter fall within the class of Groves mechanisms.

anisms that satisfy both properties [51, 65]. The VCG mechanism has an additional nice property, which is that it satisfies the *non-deficit* property (in allocation settings), which means that the mechanism does not need to be subsidized by an outside party. However, the VCG mechanism does not satisfy the non-deficit constraint—generally value flows out of the system of agents, in the form of VCG payments. In the context of auctions, often, this is not seen as a problem for the sake of maximizing the agents' welfare (the agents' total utility): the idea is that the payments are collected by the seller of the items, who is just another agent, so that nothing goes to waste. However, this reasoning does not apply to many multiagent settings; in particular, it does not apply to settings in which *there is no seller* who is separate from the agents. For example, consider the problem of dissolving a partnership: suppose there is a group of agents who have started a company together, but due to personal disagreements can no longer work together, so that it becomes essential to allocate the (currently jointly owned) company to just one of the agents. While it makes sense to auction off the company among the agents, ideally, the revenue of this auction is then distributed among the agents themselves—if the revenue leaves the system of the agents, their welfare is reduced. Similarly, the agents may be deciding how to allocate a resource that is not claimed by anyone—for example, the agents may have jointly discovered a valuable commodity (say, an oil field) in unclaimed territory, which they now need to allocate to the one of them that can make the best use of it. Finally, the agents may have a jointly owned resource (say, a powerful computer) that can only be used by one agent on any given day, and may wish to use an auction to determine which agent gets to use it today. In all these cases, any payment that is not redistributed to the agents truly goes to waste. Hence, to maximize social welfare (taking payments into account), we would prefer a budget balanced mechanism to one that merely achieves the non-deficit property (assuming both are efficient). Unfortunately, it is impossible to achieve budget balance together with strategy-proofness and efficiency [67, 51, 87].[2] Incidentally, while these types of setting are perhaps not what one typically has in mind when considering "auctions" in the common sense of the word, the fact that we use auctions does not significantly limit the generality of our approach. Effectively, we just use "auctions" as a convenient word to describe resource allocation mechanisms that use payments.

Previous research has sacrificed either strategy-proofness or efficiency to achieve budget balance [46, 90, 45]. Another approach is to allocate the items according to the VCG mechanism, and then to redistribute as much of the total VCG payment as possible back to the agents, in a way that does not affect the desirable properties of the VCG mechanism. Several papers have pursued this idea and proposed some natural redistribution mechanisms [8, 94, 20]. For example, in the Bailey mechanism [8], each agent receives a redistribution payment that equals $1/n$ times the VCG revenue that would result if this agent were removed from the auction. In the Cavallo mechanism [20], each agent receives a redistribution payment that equals $1/n$ times the minimal VCG revenue that can be obtained by changing this agent's own bid. For revenue monotonic settings,[3] Bailey's and Cavallo's mechanisms coincide; in this case we refer to this mechanism as the Bailey-Cavallo mechanism. As an example, in the special case of a single-item auction, under the Bailey-Cavallo mechanism, an agent's redistribution payment is $1/n$ times the second-highest bid among *other* agents' bids. That is, the agent with the highest bid wins and pays

[2] The dAGVA mechanism [41] is efficient, (strongly) budget balanced, and *Bayes-Nash* incentive compatible, which means that if each agent's belief over the other agents' valuations is the distribution that results from conditioning the (common) prior distribution over valuations on the agent's own valuation, and other agents bid truthfully, then the agent is best off (in expectation) bidding truthfully. In practice, it is somewhat unreasonable to assume that agents' beliefs are so consistent with each other and with the mechanism designer's belief, so we use the much stronger and more common notion of dominant-strategies incentive compatibility (strategy-proofness) in this chapter.

[3] It is well-known that for general valuations, the VCG mechanism does *not* satisfy this revenue monotonicity criterion [7, 35, 108, 110, 111] (this is in fact true for a much wider class of mechanisms [96]). However, with restricted valuations, such as the ones considered in this chapter, revenue monotonicity often holds.

the second-highest bid, as in a second-price sealed-bid (Vickrey) auction; then, the agents with the highest and the second-highest bids each receive a redistribution payment of v_3/n, where v_3 is the third-highest bid; and the remaining agents each receive a redistribution payment of v_2/n, where v_2 is the second-highest bid. Hence, the total redistributed is $2v_3/n + (n-2)v_2/n \leq nv_2/n = v_2$. That is, there is never a deficit. Other desirable properties of the VCG mechanism are also maintained by the above mechanism. For example, because an agent's redistribution does not depend on her own bid, the agents' incentives are not affected by the redistribution step. That is, the above mechanism maintains strategy-proofness.

In this chapter, we extend the idea behind the Bailey-Cavallo mechanism. We aim to design *optimal* redistribution mechanisms. In Section 2.1, we study the problem of designing VCG redistribution mechanisms that redistribute the most in the *worst case*. In Section 2.2, we study the problem of designing VCG redistribution mechanisms that redistribute the most in *expectation*. In Section 2.3, we study the problem of designing mechanisms whose redistribution functions are *undominated* in the sense that no other mechanisms can always perform as well as, and sometimes better. Finally, in Section 2.4, we study the problem of designing allocation rule together with the redistribution scheme, allowing for the allocation to be inefficient. Most of the results in this chapter are obtained based on the computationally feasible automated mechanism design approach.

2.1 Worst-Case Optimal Redistribution of VCG Payments

In this section, we study the problem of designing VCG redistribution mechanisms that redistribute the most in the *worst case*. We mainly focus on allocation settings where there are multiple indistinguishable units of a single good, and each agent's valuation function is concave—that is, agents have nonincreasing marginal values. The settings we consider here are revenue monotonic. That is, in our settings, Cavallo's mechanism and Bailey's mechanism coincide.[4]

From Subsection 2.1.1 to Subsection 2.1.8, we consider a slightly simpler setting where all agents have *unit demand*, *i.e.* they want only a single unit. We propose the family of *linear* VCG redistribution mechanisms. All mechanisms in this family are efficient, strategy-proof, individually rational, and never incur a deficit. The family includes the Bailey-Cavallo mechanism as a special case (with the caveat that Bailey's and Cavallo's mechanisms can be applied in more general settings). We then provide an optimization model for finding the optimal mechanism inside the family, based on worst-case analysis. We convert this optimization model into a linear program. Both numerical and analytical solutions of this linear program are provided, and the resulting mechanism shows significant improvement over the Bailey-Cavallo mechanism (in the worst case). For example, for the problem of allocating a single unit, when the number of agents is 10, the resulting mechanism always redistributes more than 98% of the total VCG payment back to the agents (whereas the Bailey-Cavallo mechanism redistributes only 80% in the worst case). Finally, we prove that this mechanism is in fact optimal among *all* mechanisms (even nonlinear ones) that satisfy the desirable properties.

Around the same time, the same mechanism (in the unit demand setting only)

[4] When there is only a single unit, the same mechanism has also been proposed by Porter *et al.* [94].

has been independently derived by Moulin [84].[5] Moulin actually pursues a different objective (also based on worst-case analysis): whereas our objective is to maximize the fraction of VCG payments that are redistributed, Moulin tries to minimize the overall payments from agents as a fraction of efficiency. It turns out that the resulting mechanisms are the same. However, for our objective, the optimal mechanism does not change even if the individual rationality requirement is dropped, while for Moulin's objective, dropping individual rationality does change the optimal mechanism (but only if there are multiple units).

In Subsection 2.1.8, we drop the non-deficit requirement and solve for the mechanism that is as close to budget balance as possible (in the worst case). This mechanism is in fact closer to budget balance than the best non-deficit mechanism.[6]

In Subsection 2.1.9, we consider the more general setting where the agents do not necessarily have unit demand, but have nonincreasing marginal values. We generalize the optimal redistribution mechanism to this setting (both with and without the individual rationality constraint, and both with or without the non-deficit constraint). In each case, the worst-case performance is the same as for the unit demand setting.

Finally, in Subsection 2.1.10, we consider multi-unit auctions without restrictions on the agents' valuations—marginal values may increase. Here, we show a negative result: when there are at least two units, no redistribution mechanism performs better (in the worst case) than the original VCG mechanism (redistributing nothing).

2.1.1 Formalization

From this subsection to Subsection 2.1.8, we consider only the unit demand setting. Let n denote the number of agents, and let m denote the number of units. At the

[5] We thank Rakesh Vohra for pointing us to Moulin's paper.

[6] Moulin [84] also notes that dropping the non-deficit requirement can bring us closer to budget balance, but does not solve for the optimal mechanism.

moment, we only consider the case where $m < n$ (otherwise the problem becomes trivial in the unit demand setting). We also assume that m and n are always known. This assumption is not harmful: in environments where anyone can join the auction, running a redistribution mechanism is typically not a good idea anyway, because everyone would want to join to collect part of the redistribution.

In the unit demand setting, an agent's marginal value for any unit after the first is zero. Hence, the agent's type corresponds to a single value, which is her valuation for having at least one unit.

Let the set of agents be $I = \{1, 2, \ldots, n\}$. Since in the unit demand setting, an agent's type corresponds to a single value, we simply use v_i to denote agent i's type, where v_i is agent i's valuation for having at least one unit. Without loss of generality, we assume agent i is the agent with the ith highest type v_i. That is, we have $v_1 \geq v_2 \geq \ldots \geq v_n \geq 0$. Throughout this chapter, we only consider strategy-proof mechanisms. Therefore, v_i is both agent i's true type and reported type.

Under the VCG mechanism, each agent among $1, \ldots, m$ wins a unit, and pays v_{m+1} for this unit. Thus, the total VCG payment equals mv_{m+1}. When $m = 1$, this is the second-price or Vickrey auction.

We modify the mechanism as follows. After running the original VCG mechanism, agent i receives some redistribution amount z_i (agent i's *redistribution payment*). We do not allow z_i to depend on v_i; because of this, agent i's incentives are unaffected by this redistribution payment, and the mechanism remains strategy-proof.

2.1.2 Linear VCG Redistribution Mechanisms

We are now ready to introduce the family of linear VCG redistribution mechanisms. Such a mechanism is defined by a vector of constants $c_0, c_1, \ldots, c_{n-1}$. The amount that the mechanism returns to agent i is $z_i = c_0 + c_1 v_1 + c_2 v_2 + \ldots + c_{i-1} v_{i-1} + c_i v_{i+1} + \ldots + c_{n-1} v_n$. That is, an agent receives c_0, plus c_1 times the highest bid *other* than the agent's own bid, plus c_2 times the second-highest other bid, *etc.* The mechanism is strategy-proof, because for all i, z_i is independent of v_i. Also, the mechanism is anonymous and efficient.

It is helpful to see the entire list of redistribution payments:

$$z_1 = c_0 + c_1 v_2 + c_2 v_3 + c_3 v_4 + \ldots + c_{n-2} v_{n-1} + c_{n-1} v_n$$

$$z_2 = c_0 + c_1 v_1 + c_2 v_3 + c_3 v_4 + \ldots + c_{n-2} v_{n-1} + c_{n-1} v_n$$

$$z_3 = c_0 + c_1 v_1 + c_2 v_2 + c_3 v_4 + \ldots + c_{n-2} v_{n-1} + c_{n-1} v_n$$

$$z_4 = c_0 + c_1 v_1 + c_2 v_2 + c_3 v_3 + \ldots + c_{n-2} v_{n-1} + c_{n-1} v_n$$

$$\vdots$$

$$z_i = c_0 + c_1 v_1 + c_2 v_2 + \ldots + c_{i-1} v_{i-1} + c_i v_{i+1} + \ldots + c_{n-1} v_n$$

$$\vdots$$

$$z_{n-2} = c_0 + c_1 v_1 + c_2 v_2 + c_3 v_3 + \ldots + c_{n-2} v_{n-1} + c_{n-1} v_n$$

$$z_{n-1} = c_0 + c_1 v_1 + c_2 v_2 + c_3 v_3 + \ldots + c_{n-2} v_{n-2} + c_{n-1} v_n$$

$$z_n = c_0 + c_1 v_1 + c_2 v_2 + c_3 v_3 + \ldots + c_{n-2} v_{n-2} + c_{n-1} v_{n-1}$$

Not all choices of the constants $c_0, \ldots, c_{n-1}$ produce a mechanism that is individually rational, and not all choices of the constants produce a mechanism that never incurs a deficit. Hence, to obtain these properties, we need to place some constraints on the constants.

To satisfy the individual rationality criterion, each agent's utility should always be nonnegative. An agent that does not win a unit obtains a utility that is equal to the agent's redistribution payment. An agent that wins a unit obtains a utility that is equal to the agent's valuation for the unit, minus the VCG payment v_{m+1}, plus the agent's redistribution payment.

Consider agent n, the agent with the lowest bid. Since this agent does not win an item ($m < n$), her utility is just her redistribution payment z_n. Hence, for the mechanism to be individually rational, the c_i must be such that z_n is always nonnegative.

If the c_i have this property, then it actually follows that z_i is nonnegative for *every* i, for the following reason. Suppose there exists some $i < n$ and some vector of bids $v_1 \geq v_2 \geq \ldots \geq v_n \geq 0$ such that $z_i < 0$. Then, consider the bid vector that results from replacing v_j by v_{j+1} for all $j \geq i$, and letting $v_n = 0$. If we omit v_n from this vector, the same vector results that results from omitting v_i from the original vector. Therefore, n's redistribution payment under the new vector should be the same as i's redistribution payment under the old vector—but this payment is negative.

If all redistribution payments are always nonnegative, then the mechanism must be individually rational (because the VCG mechanism is individually rational, and the redistribution payment only increases an agent's utility). Therefore, the mechanism is individually rational if and only if for any bid vector, $z_n \geq 0$.

To satisfy the non-deficit criterion, the sum of the redistribution payments should be less than or equal to the total VCG payment. So for any bid vector $v_1 \geq v_2 \geq \ldots \geq v_n \geq 0$, the constants c_i should make $z_1 + z_2 + \ldots + z_n \leq mv_{m+1}$.

We will focus on linear VCG redistribution mechanisms that satisfy both the individual rationality and the non-deficit property. That is, we require that the c_i satisfy the above two constraints. It turns out that some of the c_i always need to be set to 0, as the following proposition demonstrates.

Proposition 1. *If $c_0, c_1, \ldots, c_{n-1}$ satisfy both the individual rationality and the non-deficit constraints, then $c_i = 0$ for $i = 0, \ldots, m$.*

Proof. First, let us prove that $c_0 = 0$. Consider the bid vector in which $v_i = 0$ for all i. To obtain individual rationality, we must have $c_0 \geq 0$. To satisfy the non-deficit constraint, we must have $c_0 \leq 0$. Thus we know $c_0 = 0$. Now, if $c_i = 0$ for all i, there is nothing to prove. Otherwise, let $j = min\{i|c_i \neq 0\}$. Assume that $j \leq m$. We recall that we can write the individual rationality constraint as follows: $z_n = c_0 + c_1v_1 + c_2v_2 + c_3v_3 + \ldots + c_{n-2}v_{n-2} + c_{n-1}v_{n-1} \geq 0$ for any bid vector. Let us consider the bid vector in which $v_i = 1$ for $i \leq j$ and $v_i = 0$ for the rest. In this case $z_n = c_j$, so we must have $c_j \geq 0$. The non-deficit constraint can be written as follows: $z_1 + z_2 + \ldots + z_n \leq m v_{m+1}$ for any bid vector. Consider the same bid vector as above. We have $z_i = 0$ for $i \leq j$, because for these bids, the jth highest other bid has value 0, so all the c_i that are nonzero are multiplied by 0. For $i > j$, we have $z_i = c_j$, because the jth highest other bid has value 1, and all lower bids have value 0. So the non-deficit constraint tells us that $c_j(n - j) \leq m v_{m+1}$. Because $j \leq m$, $v_{m+1} = 0$, so the right hand side is 0. We also have $n - j > 0$ because $j \leq m < n$. So $c_j \leq 0$. Because we have already established that $c_j \geq 0$, it follows that $c_j = 0$; but this is contrary to assumption. So $j > m$. $\square$

Incidentally, this proposition also shows that if $m = n - 1$, then $c_i = 0$ for all i. Thus, we are stuck with the VCG mechanism (more details in Proposition 7). From here on, we only consider the case where $m < n - 1$.

We now give two examples of mechanisms in this family.

Example 1. Bailey-Cavallo mechanism: Consider the mechanism corresponding to $c_{m+1} = \frac{m}{n}$ and $c_i = 0$ for all other i. Under this mechanism, each agent receives a redistribution payment of $\frac{m}{n}$ times the $(m+1)$th highest bid from another agent. Hence, $1, \ldots, m+1$ receive a redistribution payment of $\frac{m}{n} v_{m+2}$, and the others receive $\frac{m}{n} v_{m+1}$. Thus, the total redistribution payment is $(m+1)\frac{m}{n} v_{m+2} + (n-m-1)\frac{m}{n} v_{m+1}$. This redistribution mechanism is individually rational, because all the redistribution payments are nonnegative, and never incurs a deficit, because $(m+1)\frac{m}{n} v_{m+2} + (n - m - 1)\frac{m}{n} v_{m+1} \leq n \frac{m}{n} v_{m+1} = m v_{m+1}$. (We note that for this mechanism to make sense, we need $n \geq m + 2$.)

Example 2. Consider the mechanism corresponding to $c_{m+1} = \frac{m}{n-m-1}$, $c_{m+2} = -\frac{m(m+1)}{(n-m-1)(n-m-2)}$, and $c_i = 0$ for all other i. In this mechanism, each agent receives a redistribution payment of $\frac{m}{n-m-1}$ times the $(m+1)$th highest reported value from other agents, minus $\frac{m(m+1)}{(n-m-1)(n-m-2)}$ times the $(m+2)$th highest reported value from other agents. Thus, the total redistribution payment is $mv_{m+1} - \frac{m(m+1)(m+2)}{(n-m-1)(n-m-2)}v_{m+3}$. If $n \geq 2m+3$ (which is equivalent to $\frac{m}{n-m-1} \geq \frac{m(m+1)}{(n-m-1)(n-m-2)}$), then each agent always receives a nonnegative redistribution payment, thus the mechanism is individually rational. Also, the mechanism never incurs a deficit, because the total VCG payment is mv_{m+1}, which is greater than the amount $mv_{m+1} - \frac{m(m+1)(m+2)}{(n-m-1)(n-m-2)}v_{m+3}$ that is redistributed.

Which of these two mechanisms is better? Is there another mechanism that is even better? This is what we study in the next subsection.

2.1.3 Worst-case Optimal Redistribution Mechanisms

Among all linear VCG redistribution mechanisms, we would like to be able to identify the one that redistributes the greatest fraction of the total VCG payment.[7] This is not a well-defined notion: it may be that one mechanism redistributes more on some bid vectors, and another more on other bid vectors. In this section, we compare mechanisms by the fraction of the total VCG payment that they redistribute in the worst case. This fraction is undefined when the total VCG payment is 0. To deal with this, technically, we define the worst-case redistribution fraction as the largest k so that the total amount redistributed is at least k times the total VCG payment, for all bid vectors. (Hence, as long as the total amount redistributed is at least 0 when the total VCG payment is 0, these cases do not affect the worst-case fraction.)

Let us analyze the worst-case performances of the two example mechanisms mentioned earlier. For the first example, the total redistribution payment is $(m +$

[7] The fraction redistributed seems a natural criterion to use. One good property of this criterion is that it is scale-invariant: if we multiply all bids by the same positive constant (for example, if we change the units by re-expressing the bids in euros instead of dollars), we would not want the behavior of our mechanism to change.

$1)\frac{m}{n}v_{m+2}+(n-m-1)\frac{m}{n}v_{m+1}$, which is greater than or equal to $(n-m-1)\frac{m}{n}v_{m+1}$. In the worst case, which is when $v_{m+2}=0$, the fraction redistributed is $\frac{n-m-1}{n}$. For the second example, the total redistribution payment is $mv_{m+1}-\frac{m(m+1)(m+2)}{(n-m-1)(n-m-2)}v_{m+3}$, which is greater than or equal to $mv_{m+1}(1-\frac{(m+1)(m+2)}{(n-m-1)(n-m-2)})$. In the worst case, which is when $v_{m+3}=v_{m+1}$, the fraction redistributed is $1-\frac{(m+1)(m+2)}{(n-m-1)(n-m-2)}$. Since we assume that the number of agents n and the number of units m are known, we can determine which example mechanism has better worst-case performance by comparing the two quantities. When $n=6$ and $m=1$, for the first example (Bailey-Cavallo mechanism), the fraction redistributed in the worst case is $\frac{2}{3}$, and for the second example, this fraction is $\frac{1}{2}$, which implies that for this pair of n and m, the first mechanism has better worst-case performance. On the other hand, when $n=12$ and $m=1$, for the first example, the fraction redistributed in the worst case is $\frac{5}{6}$, and for the second example, this fraction is $\frac{14}{15}$, which implies that this time the second mechanism has better worst-case performance.

The problem of finding the worst-case optimal VCG redistribution mechanism corresponds to the following optimization problem:

Maximize k (the fraction redistributed in the worst case)
Subject to:
For every bid vector $v_1 \geq v_2 \geq \ldots \geq v_n \geq 0$
$z_n \geq 0$ (individual rationality)
$z_1+z_2+\ldots+z_n \leq mv_{m+1}$ (non-deficit)
$z_1+z_2+\ldots+z_n \geq kmv_{m+1}$ (worst-case constraint)
We recall that $z_i=c_0+c_1v_1+c_2v_2+\ldots+c_{i-1}v_{i-1}+c_iv_{i+1}+\ldots+c_{n-1}v_n$

2.1.4 Transformation to Linear Programming

The optimization problem given in the previous subsection can be rewritten as a linear program, based on the following observations.

Proposition 2. *The individual rationality constraint can be written as follows:* $\sum_{i=m+1}^{j} c_i \geq 0$ *for* $j=m+1,\ldots,n-1$.

Before proving this proposition, we introduce the following lemma.

Lemma 1. *Given a positive integer k and a set of real constants $s_1, s_2, \ldots, s_k$, ($s_1t_1 + s_2t_2 + \ldots + s_kt_k \geq 0$ for any $t_1 \geq t_2 \geq \ldots \geq t_k \geq 0$) if and only if ($\sum_{i=1}^{j} s_i \geq 0$ for $j = 1, 2, \ldots, k$).*

Proof. Let $d_i = t_i - t_{i+1}$ for $i = 1, 2, \ldots, k-1$, and $d_k = t_k$. Then ($s_1t_1 + s_2t_2 + \ldots + s_kt_k \geq 0$ for any $t_1 \geq t_2 \geq \ldots \geq t_k \geq 0$) is equivalent to $((\sum_{i=1}^{1} s_i)d_1 + (\sum_{i=1}^{2} s_i)d_2 + \ldots + (\sum_{i=1}^{k} s_i)d_k \geq 0$ for any set of arbitrary nonnegative d_i). When $\sum_{i=1}^{j} s_i \geq 0$ for $j = 1, 2, \ldots, k$, the above inequality is obviously true. If for some j, $\sum_{i=1}^{j} s_i < 0$, if we set $d_j > 0$ and $d_i = 0$ for all $i \neq j$, then the above inequality becomes false. So $\sum_{i=1}^{j} s_i \geq 0$ for $j = 1, 2, \ldots, k$ is both necessary and sufficient. □

We are now ready to present the proof of Proposition 2.

Proof. The individual rationality constraint can be written as $z_n = c_0 + c_1v_1 + c_2v_2 + c_3v_3 + \ldots + c_{n-2}v_{n-2} + c_{n-1}v_{n-1} \geq 0$ for any bid vector $v_1 \geq v_2 \geq \ldots \geq v_{n-1} \geq v_n \geq 0$. We have already shown that $c_i = 0$ for $i \leq m$. Thus, the above can be simplified to $z_n = c_{m+1}v_{m+1} + c_{m+2}v_{m+2} + \ldots + c_{n-2}v_{n-2} + c_{n-1}v_{n-1} \geq 0$ for any bid vector. By the above lemma, this is equivalent to $\sum_{i=m+1}^{j} c_i \geq 0$ for $j = m+1, \ldots, n-1$. □

Proposition 3. *The non-deficit constraint and the worst-case constraint can also be written as linear inequalities involving only the c_i and k.*

Proof. The non-deficit constraint requires that for any bid vector, $z_1 + z_2 + \ldots + z_n \leq mv_{m+1}$, where $z_i = c_0 + c_1v_1 + c_2v_2 + \ldots + c_{i-1}v_{i-1} + c_iv_{i+1} + \ldots + c_{n-1}v_n$ for $i = 1, 2, \ldots, n$. Because $c_i = 0$ for $i \leq m$, we can simplify this inequality to

$q_{m+1}v_{m+1} + q_{m+2}v_{m+2} + \ldots + q_nv_n \geq 0$

$q_{m+1} = m - (n-m-1)c_{m+1}$

$q_i = -(i-1)c_{i-1} - (n-i)c_i$, for $i = m+2, \ldots, n-1$ (when $m+2 > n-1$, this set of equalities is empty)

$q_n = -(n-1)c_{n-1}$

By the above lemma, this is equivalent to $\sum_{i=m+1}^{j} q_i \geq 0$ for $j = m+1, \ldots, n$. So, we can simplify further as follows:

$q_{m+1} \geq 0 \iff (n-m-1)c_{m+1} \leq m$

$q_{m+1} + \ldots + q_{m+i} \geq 0 \iff n\sum_{j=m+1}^{j=m+i-1} c_j + (n-m-i)c_{m+i} \leq m$ for $i = 2, \ldots, n-m-1$

$q_{m+1} + \ldots + q_n \geq 0 \iff n\sum_{j=m+1}^{j=n-1} c_j \leq m$

So, the non-deficit constraint can be written as a set of linear inequalities involving only the c_i.

The worst-case constraint can be also written as a set of linear inequalities, by the following reasoning. The worst-case constraint requires that for any bid input $z_1 + z_2 + \ldots + z_n \geq kmv_{m+1}$, where $z_i = c_0 + c_1v_1 + c_2v_2 + \ldots + c_{i-1}v_{i-1} + c_iv_{i+1} + \ldots + c_{n-1}v_n$ for $i = 1, 2, \ldots, n$. Because $c_i = 0$ for $i \leq m$, we can simplify this inequality to

$Q_{m+1}v_{m+1} + Q_{m+2}v_{m+2} + \ldots + Q_nv_n \geq 0$

$Q_{m+1} = (n-m-1)c_{m+1} - km$

$Q_i = (i-1)c_{i-1} + (n-i)c_i$, for $i = m+2, \ldots, n-1$

$Q_n = (n-1)c_{n-1}$

By the above lemma, this is equivalent to $\sum_{i=m+1}^{j} Q_i \geq 0$ for $j = m+1, \ldots, n$. So, we can simplify further as follows:

$Q_{m+1} \geq 0 \iff (n-m-1)c_{m+1} \geq km$

$Q_{m+1} + \ldots + Q_{m+i} \geq 0 \iff n\sum_{j=m+1}^{j=m+i-1} c_j + (n-m-i)c_{m+i} \geq km$ for $i = 2, \ldots, n-m-1$

$Q_{m+1} + \ldots + Q_n \geq 0 \iff n\sum_{j=m+1}^{j=n-1} c_j \geq km$

So, the worst-case constraint can also be written as a set of linear inequalities involving only the c_i and k. □

Combining all the propositions, we see that the original optimization problem can be transformed into the following linear program. It should be noted that we need different linear programs for different n and m.

Variables: $c_{m+1}, c_{m+2}, \ldots, c_{n-1}, k$
Maximize k (the fraction redistributed in the worst case)
Subject to:
$\sum_{i=m+1}^{j} c_i \geq 0$ for $j = m+1, \ldots, n-1$
$km \leq (n-m-1)c_{m+1} \leq m$
$km \leq n\sum_{j=m+1}^{j=m+i-1} c_j + (n-m-i)c_{m+i} \leq m$ for $i = 2, \ldots, n-m-1$
$km \leq n\sum_{j=m+1}^{j=n-1} c_j \leq m$

2.1.5 Numerical Results

For selected values of n and m, we solved the linear program using GLPK (GNU Linear Programming Kit). In this subsection, we compare the resulting mechanisms with the Bailey-Cavallo mechanism.

Worst-case performance

In Table 2.1, we present the results for a single unit ($m = 1$). The second column displays the fraction of the total VCG payment that is not redistributed in the worst case by the worst-case optimal mechanism—that is, it displays the value $1 - k$. (Displaying k would require too many significant digits.) Correspondingly, the third column displays the fraction of the total VCG payment that is not redistributed by the Bailey-Cavallo mechanism in the worst case (which is equal to $\frac{2}{n}$).

In Table 2.1, we showed that when $m = 1$, the worst-case optimal mechanism significantly outperforms the Bailey-Cavallo mechanism in the worst case. For larger m ($m = 1, 2, 3, 4$, $n = m+2, \ldots, 30$), we compare the worst-case performance of these two mechanisms in Figure 2.1. We see that for any m, when $n = m + 2$, the worst-case optimal mechanism has the same worst-case performance as the Bailey-Cavallo mechanism (actually, in this case, the worst-case optimal mechanism is identical to

Table 2.1: Worst-case performances under the WCO and the Bailey-Cavallo mechanisms for single-item case.

n	**Worst-Case Optimal Mechanism**	**Bailey-Cavallo Mechanism**
3	66.7%	66.7%
4	42.9%	50.0%
5	26.7%	40.0%
6	16.1%	33.3%
7	9.52%	28.6%
8	5.51%	25.0%
9	3.14%	22.2%
10	1.76%	20.0%
15	8.55e-4	13.3%
20	3.62e-5	10.0%
30	5.40e-8	6.67e-2
40	7.09e-11	5.00e-2

the Bailey-Cavallo mechanism). When $n > m+2$, the worst-case optimal mechanism outperforms the Bailey-Cavallo mechanism (in the worst case).

In Subsection 2.1.9, we will see that in the more general setting where agents have nonincreasing marginal values, the worst-case redistribution fraction for the (generalized) worst-case optimal mechanism is the same as for the unit demand setting. The same is true for the Bailey-Cavallo mechanism. Hence, Figure 2.1 does not change in this more general setting.

Average-case performance

It is perhaps not surprising that the worst-case optimal mechanism significantly outperforms the Bailey-Cavallo mechanism in the worst case, because that is, after all, the case for which the former has been designed. We can also compare how much the mechanisms redistribute on average (say, when the bids are drawn i.i.d. from a uniform distribution over $[0, 1]$). In this case, the worst-case optimal mechanism does not always outperform the Bailey-Cavallo mechanism. Table 2.2 compares the expected amount of VCG payment that fails to be redistributed by the worst-case

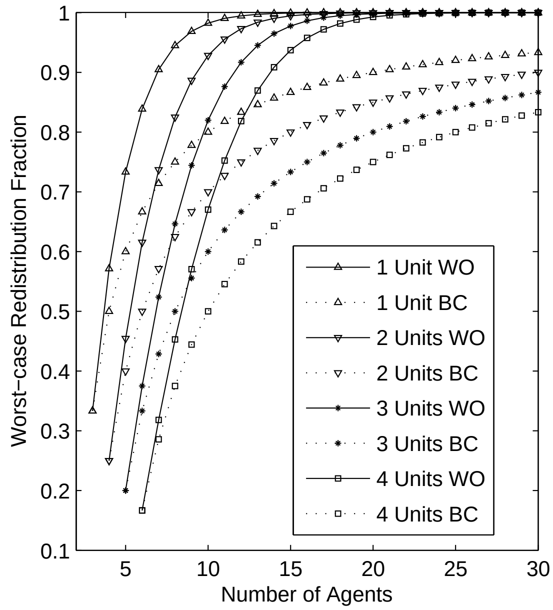

FIGURE 2.1: A comparison of the worst-case performance of the worst-case optimal mechanism (WCO) and the Bailey-Cavallo mechanism (BC).

optimal mechanism and by the Bailey-Cavallo mechanism ($m = 1$).

We see that when n is small, the Bailey-Cavallo mechanism outperforms the worst-case optimal redistribution mechanism in expectation (except for the case $n = 3$, for which the two mechanisms are the same). When n is large ($n \geq 8$), the worst-case optimal redistribution mechanism outperforms the Bailey-Cavallo mechanism. The results are similar for larger m. That is, when n is small, the Bailey-Cavallo mechanism outperforms the worst-case optimal redistribution mechanism in expectation (except for the case $n = m + 2$, for which the two mechanisms are the same). When n is large (*e.g.* $n \geq 10$ for $m = 2$; $n \geq 13$ for $m = 3$; $n \geq 16$ for $m = 4$), the worst-case optimal redistribution mechanism performs better than the Bailey-Cavallo mechanism. In fact, this is not surprising: the expected amount that fails to be redistributed by the Bailey-Cavallo mechanism vanishes as $\Theta(\frac{1}{n^2})$. This is slower than the convergence rate of the *worst-case* redistribution fraction for the worst-case optimal mechanism (Corollary 1); and, of course, the average-case performance of

Table 2.2: Average-case performances under the WCO and the Bailey-Cavallo mechanisms for single-item case.

n	**Worst-Case Optimal Mechanism**	**Bailey-Cavallo Mechanism**
3	0.1667	0.1667
4	0.1714	0.1000
5	0.08889	0.06667
6	0.06912	0.04762
7	0.03571	0.03571
8	0.02450	0.02778
9	0.01255	0.02222
10	0.008006	0.01818
15	3.739e-4	0.008333
20	1.726e-5	0.004762
30	2.614e-8	0.002151
40	3.461e-11	0.001220

the worst-case optimal mechanism must be at least as good as its worst-case performance. This also shows that the worst-case optimal mechanism asymptotically outperforms the Bailey-Cavallo mechanism, even in the average case.

A detailed example

Finally, let us present the result for the case $n = 5, m = 1$ in detail. By solving the above linear program, we find that the optimal values for the c_i are $c_2 = \frac{11}{45}, c_3 = -\frac{1}{9}$, and $c_4 = \frac{1}{15}$. That is, the redistribution payment received by each agent under the worst-case optimal mechanism is: $\frac{11}{45}$ times the second highest bid among the other agents, minus $\frac{1}{9}$ times the third highest bid among the other agents, plus $\frac{1}{15}$ times the fourth highest bid among the other agents.

$$\text{agent 1 receives } \tfrac{11}{45}v_3 - \tfrac{1}{9}v_4 + \tfrac{1}{15}v_5$$
$$\text{agent 2 receives } \tfrac{11}{45}v_3 - \tfrac{1}{9}v_4 + \tfrac{1}{15}v_5$$
$$\text{agent 3 receives } \tfrac{11}{45}v_2 - \tfrac{1}{9}v_4 + \tfrac{1}{15}v_5$$
$$\text{agent 4 receives } \tfrac{11}{45}v_2 - \tfrac{1}{9}v_3 + \tfrac{1}{15}v_5$$
$$\text{agent 5 receives } \tfrac{11}{45}v_2 - \tfrac{1}{9}v_3 + \tfrac{1}{15}v_4$$

The total amount redistributed by the worst-case optimal mechanism is $\frac{11}{15}v_2 + \frac{4}{15}v_3 - \frac{4}{15}v_4 + \frac{4}{15}v_5$; in the worst case, $\frac{11}{15}v_2$ is redistributed. Hence, the fraction of the total VCG payment that is not redistributed is never more than $\frac{4}{15} = 26.7\%$.

As a specific example, for the bid vector $v_1 = 4, v_2 = 3, v_3 = 2, v_4 = 1, v_5 = 1$, the total amount redistributed by the worst-case optimal redistribution mechanism is $\frac{11}{15}v_2 + \frac{4}{15}v_3 - \frac{4}{15}v_4 + \frac{4}{15}v_5 = \frac{11}{15}3 + \frac{4}{15}2 - \frac{4}{15}1 + \frac{4}{15}1 = \frac{41}{15}$. The total amount redistributed by the Bailey-Cavallo mechanism is $\frac{2}{5}v_3 + \frac{3}{5}v_2 = \frac{2}{5}2 + \frac{3}{5}3 = \frac{13}{5}$. Hence, for this bid vector, the worst-case optimal redistribution mechanism redistributes more.

As another specific example, for the bid vector $v_1 = 4, v_2 = 3, v_3 = 2, v_4 = 2, v_5 = 1$, the total amount redistributed by the worst-case optimal redistribution mechanism is $\frac{11}{15}v_2 + \frac{4}{15}v_3 - \frac{4}{15}v_4 + \frac{4}{15}v_5 = \frac{11}{15}3 + \frac{4}{15}2 - \frac{4}{15}2 + \frac{4}{15}1 = \frac{37}{15}$. The total amount redistributed by the Bailey-Cavallo mechanism is still $\frac{13}{5}$. Hence, for this bid vector, the Bailey-Cavallo mechanism redistributes more.

2.1.6 Analytical Characterization

We recall that our linear program has the following form:

Variables: $c_{m+1}, c_{m+2}, \ldots, c_{n-1}, k$
Maximize k (the fraction redistributed in the worst case)
Subject to:
$\sum_{i=m+1}^{j} c_i \geq 0$ for $j = m+1, \ldots, n-1$
$km \leq (n-m-1)c_{m+1} \leq m$
$km \leq n\sum_{j=m+1}^{j=m+i-1} c_j + (n-m-i)c_{m+i} \leq m$ for $i = 2, \ldots, n-m-1$
$km \leq n\sum_{j=m+1}^{j=n-1} c_j \leq m$

A linear program has no solution if and only if either the objective is unbounded, or the constraints are contradictory (there is no feasible solution). It is easy to see that k is bounded above by 1 (redistributing more than 100% violates the non-deficit constraint). Also, a feasible solution always exists, for example, $k = 0$ and $c_i = 0$ for all i. So an optimal solution always exists. Observe that the linear program model depends only on the number of agents n and the number of units m. Hence the optimal solution is a function of n and m. It turns out that this optimal solution can be analytically characterized as follows.

Theorem 1. *For any m and n with $n \geq m+2$, the worst-case optimal mechanism (among linear VCG redistribution mechanisms) is unique. For this mechanism, the fraction redistributed in the worst case is*

$$k^* = 1 - \frac{\binom{n-1}{m}}{\sum_{j=m}^{n-1} \binom{n-1}{j}}$$

The worst-case optimal mechanism is characterized by the following values for the c_i:

$$c_i^* = \frac{(-1)^{i+m-1}(n-m)\binom{n-1}{m-1}}{i \sum_{j=m}^{n-1} \binom{n-1}{j}} \frac{1}{\binom{n-1}{i}} \sum_{j=i}^{n-1} \binom{n-1}{j}$$

for $i = m+1, \ldots, n-1$.

It should be noted that we have proved $c_i = 0$ for $i \leq m$ in Proposition 1.

Proof. We first rewrite the linear program as follows. We introduce new variables $x_{m+1}, x_{m+2}, \ldots, x_{n-1}$, defined by $x_j = \sum_{i=m+1}^{j} c_i$ for $j = m+1, \ldots, n-1$. The linear program then becomes:

Variables: $x_{m+1}, x_{m+2}, \ldots, x_{n-1}, k$
Maximize k
Subject to:
$km \leq (n-m-1)x_{m+1} \leq m$
$km \leq (m+i)x_{m+i-1} + (n-m-i)x_{m+i} \leq m$ for $i = 2, \ldots, n-m-1$
$km \leq nx_{n-1} \leq m$
$x_i \geq 0$ for $i = m+1, m+2, \ldots, n-1$

We will prove that for any optimal solution to this linear program, $k = k^*$. Moreover, we will prove that when $k = k^*$, $x_j = \sum_{i=m+1}^{j} c_i^*$ for $j = m+1, \ldots, n-1$. This will prove the theorem.

We first make the following observations:

$(n-m-1)c^*_{m+1}$

$= (n-m-1)\frac{(n-m)\binom{n-1}{m-1}}{(m+1)\sum_{j=m}^{n-1}\binom{n-1}{j}}\frac{1}{\binom{n-1}{m+1}}\sum_{j=m+1}^{n-1}\binom{n-1}{j}$

$= (n-m-1)\frac{(n-m)\binom{n-1}{m-1}}{(m+1)\sum_{j=m}^{n-1}\binom{n-1}{j}}\frac{1}{\binom{n-1}{m+1}}(\sum_{j=m}^{n-1}\binom{n-1}{j} - \binom{n-1}{m})$

$= (n-m-1)\frac{m}{n-m-1} - (n-m-1)\frac{m\binom{n-1}{m}}{(n-m-1)\sum_{j=m}^{n-1}\binom{n-1}{j}}$

$= m - (1-k^*)m = k^*m$

For $i = m+1, \ldots, n-2$,

$ic^*_i + (n-i-1)c^*_{i+1}$

$= i\frac{(-1)^{i+m-1}(n-m)\binom{n-1}{m-1}}{i\sum_{j=m}^{n-1}\binom{n-1}{j}}\frac{1}{\binom{n-1}{i}}\sum_{j=i}^{n-1}\binom{n-1}{j}$

$+ (n-i-1)\frac{(-1)^{i+m}(n-m)\binom{n-1}{m-1}}{(i+1)\sum_{j=m}^{n-1}\binom{n-1}{j}}\frac{1}{\binom{n-1}{i+1}}\sum_{j=i+1}^{n-1}\binom{n-1}{j}$

$= \frac{(-1)^{i+m-1}(n-m)\binom{n-1}{m-1}}{\sum_{j=m}^{n-1}\binom{n-1}{j}}\frac{1}{\binom{n-1}{i}}\sum_{j=i}^{n-1}\binom{n-1}{j}$

$- (n-i-1)\frac{(-1)^{i+m-1}(n-m)\binom{n-1}{m-1}}{(i+1)\sum_{j=m}^{n-1}\binom{n-1}{j}}\frac{i+1}{\binom{n-1}{i}(n-i-1)}\sum_{j=i+1}^{n-1}\binom{n-1}{j}$

$= \frac{(-1)^{i+m-1}(n-m)\binom{n-1}{m-1}}{\sum_{j=m}^{n-1}\binom{n-1}{j}}$

$= (-1)^{i+m-1}m(1-k^*)$

Finally,

$(n-1)c^*_{n-1}$

$= (n-1)\frac{(-1)^{n+m}(n-m)\binom{n-1}{m-1}}{(n-1)\sum_{j=m}^{n-1}\binom{n-1}{j}}\frac{1}{\binom{n-1}{n-1}}\sum_{j=n-1}^{n-1}\binom{n-1}{j}$

$= (-1)^{m+n}m(1-k^*)$

Summarizing the above, we have:

$(n-m-1)c^*_{m+1} = k^*m$

$(m+1)c^*_{m+1} + (n-m-2)c^*_{m+2} = m(1-k^*)$

$(m+2)c^*_{m+2} + (n-m-3)c^*_{m+3} = -m(1-k^*)$

$(m+3)c^*_{m+3} + (n-m-4)c^*_{m+4} = m(1-k^*)$

$\vdots$

$(n-3)c^*_{n-3} + 2c^*_{n-2} = (-1)^{m+n-2}m(1-k^*)$

$(n-2)c^*_{n-2} + c^*_{n-1} = (-1)^{m+n-1}m(1-k^*)$

$(n-1)c^*_{n-1} = (-1)^{m+n}m(1-k^*)$

Let $x^*_j = \sum_{i=m+1}^{j} c^*_i$ for $j = m+1, m+2, \ldots, n-1$, the first equation in the above tells us that $(n-m-1)x^*_{m+1} = k^*m$.

By adding the first two equations, we get $(m+2)x^*_{m+1} + (n-m-2)x^*_{m+2} = m$.

By adding the first three equations, we get $(m+3)x^*_{m+2} + (n-m-3)x^*_{m+3} = k^*m$.

By adding the first i equations, where $i = 2, \ldots, n-m-1$, we get $(m+i)x^*_{m+i-1} + (n-m-i)x^*_{m+i} = m$ if i is even; $(m+i)x^*_{m+i-1} + (n-m-i)x^*_{m+i} = k^*m$ if i is odd.

Finally by adding all the equations, we get $nx^*_{n-1} = m$ if $n-m$ is even; $nx^*_{n-1} = k^*m$ if $n-m$ is odd.

Thus, for all of the constraints other than the nonnegativity constraints, we have shown that they are satisfied by setting $x_j = x^*_j = \sum_{i=m+1}^{j} c^*_i$ and $k = k^*$. We next show that the nonnegativity constraints are satisfied by these settings as well.

For $m+1 \le i, i+1 \le n-1$, we have $\frac{1}{i}\frac{\sum_{j=i}^{n-1}\binom{n-1}{j}}{\binom{n-1}{i}} = \frac{1}{i}\sum_{j=i}^{n-1}\frac{i!(n-1-i)!}{j!(n-1-j)!} \ge \frac{1}{i+1}\sum_{j=i}^{n-2}\frac{i!(n-1-i)!}{j!(n-1-j)!} \ge \frac{1}{i+1}\sum_{j=i}^{n-2}\frac{(i+1)!(n-1-i-1)!}{(j+1)!(n-1-j-1)!} = \frac{1}{i+1}\frac{\sum_{j=i+1}^{n-1}\binom{n-1}{j}}{\binom{n-1}{i+1}}$

This implies that the absolute value of c^*_i is decreasing as i increases (if the c^*_i

contains more than one number). We further observe that the sign of c_i^* alternates, with the first element c_{m+1}^* positive. So $x_j^* = \sum_{i=m+1}^{j} c_i^* \geq 0$ for all j. Thus, we have shown that these $x_i = x_i^*$ together with $k = k^*$ form a feasible solution of the linear program. We proceed to show that it is in fact the unique optimal solution.

First we prove the following proposition:

Proposition 4. *If $\hat{k}, \hat{x}_i$, $i = m+1, m+2, \ldots, n-1$ satisfy the following inequalities:*

$$\hat{k}m \leq (n-m-1)\hat{x}_{m+1} \leq m$$

$$\hat{k}m \leq (m+i)\hat{x}_{m+i-1} + (n-m-i)\hat{x}_{m+i} \leq m \text{ for } i = 2, \ldots, n-m-1$$

$$\hat{k}m \leq n\hat{x}_{n-1} \leq m$$

$$\hat{k} \geq k^*$$

then we must have that $\hat{x}_i = x_i^*$ and $\hat{k} = k^*$.

Proof of proposition. Consider the first inequality. We know that $(n-m-1)x_{m+1}^* = k^*m$, so $(n-m-1)\hat{x}_{m+1} \geq \hat{k}m \geq k^*m = (n-m-1)x_{m+1}^*$. It follows that $\hat{x}_{m+1} \geq x_{m+1}^*$ $(n-m-1 \neq 0)$.

Now, consider the next inequality for $i = 2$. We know that $(m+2)x_{m+1}^* + (n-m-2)x_{m+2}^* = m$. It follows that $(n-m-2)\hat{x}_{m+2} \leq m-(m+2)\hat{x}_{m+1} \leq m-(m+2)x_{m+1}^* = (n-m-2)x_{m+2}^*$, so $\hat{x}_{m+2} \leq x_{m+2}^*$ $(i = 2 \leq n-m-1 \Rightarrow n-m-2 \neq 0)$.

Now consider the next inequality for $i = 3$. We know that $(m+3)x_{m+2}^* + (n-m-3)x_{m+3}^* = m$. It follows that $(n-m-3)\hat{x}_{m+3} \geq \hat{k}m-(m+3)\hat{x}_{m+2} \geq k^*m-(m+3)x_{m+2}^* = (n-m-3)x_{m+3}^*$, so $\hat{x}_{m+3} \geq x_{m+3}^*$ $(i = 3 \leq n-m-1 \Rightarrow n-m-3 \neq 0)$.

Proceeding like this all the way up to $i = n-m-1$, we get that $\hat{x}_{m+i} \geq x_{m+i}^*$ if i is odd and $\hat{x}_{m+i} \leq x_{m+i}^*$ if i is even. Moreover, if one inequality is strict, then all subsequent inequalities are strict. Now, if we can prove $\hat{x}_{n-1} = x_{n-1}^*$, it would follow that the x_i^* are equal to the $\hat{x}_i$ (which also implies that $\hat{k} = k^*$).

We consider two cases:

Case 1: $n-m$ is even. We have: $n-m$ even $\Rightarrow n-m-1$ odd $\Rightarrow \hat{x}_{n-1} \geq x^*_{n-1}$. We also have: $n-m$ even $\Rightarrow nx^*_{n-1} = m$. Combining these two, we get $m = nx^*_{n-1} \leq n\hat{x}_{n-1} \leq m \Rightarrow \hat{x}_{n-1} = x^*_{n-1}$.

Case 2: $n-m$ is odd. In this case, we have $\hat{x}_{n-1} \leq x^*_{n-1}$, and $nx^*_{n-1} = k^*m$. Then, we have: $k^*m \leq \hat{k}m \leq n\hat{x}_{n-1} \leq nx^*_{n-1} = k^*m \Rightarrow \hat{x}_{n-1} = x^*_{n-1}$.

This completes the proof of the proposition. □

It follows that if $\hat{k}, \hat{x}_i$, $i = m+1, m+2, \ldots, n-1$ is a feasible solution and $\hat{k} \geq k^*$, then since all the inequalities in Proposition 4 are satisfied, we must have $\hat{x}_i = x^*_i$ and $\hat{k} = k^*$. Hence no other feasible solution is as good as the one described in the theorem. □

Knowing the analytical characterization of the worst-case optimal mechanism provides us with at least two major benefits. First, using these formulas is computationally more efficient than solving the linear program using a general-purpose solver. Second, we can derive the following corollary.

Corollary 1. *If the number of units m is fixed, then as the number of agents n increases, the worst-case fraction redistributed linearly converges to 1, with a rate of convergence $\frac{1}{2}$. (That is, $\lim_{n\to\infty} \frac{1-k^*_{n+1}}{1-k^*_n} = \frac{1}{2}$. That is, in the limit, the fraction that is not redistributed halves for every additional agent.)*

We note that this is consistent with the experimental data for the single-unit case, where the worst-case remaining fraction roughly halves each time we add another agent. The worst-case fraction that is redistributed under the Bailey-Cavallo mechanism also converges to 1 as the number of agents goes to infinity, but the convergence is much slower—it does not converge linearly (that is, letting k^C_n be the fraction redistributed by the Bailey-Cavallo mechanism in the worst case for n agents,

$\lim_{n\to\infty} \frac{1-k_{n+1}^C}{1-k_n^C} = lim_{n\to\infty} \frac{n}{n+1} = 1$). We now present the proof of the corollary.

Proof. When the number of agents is n, the worst-case fraction redistributed is $k_n^* = 1 - \frac{\binom{n-1}{m}}{\sum_{j=m}^{n-1}\binom{n-1}{j}}$. When the number of agents is $n+1$, the fraction becomes $k_{n+1}^* = 1 - \frac{\binom{n}{m}}{\sum_{j=m}^{n}\binom{n}{j}}$. For n sufficiently large, we will have $2^n - mn^{m-1} > 0$, and hence $\frac{1-k_{n+1}^*}{1-k_n^*} = \frac{\binom{n}{m}\sum_{j=m}^{n-1}\binom{n-1}{j}}{\binom{n-1}{m}\sum_{j=m}^{n}\binom{n}{j}} = \frac{n}{n-m}\frac{2^{n-1}-\sum_{j=0}^{m-1}\binom{n-1}{j}}{2^n-\sum_{j=0}^{m-1}\binom{n}{j}}$, and $\frac{n}{n-m}\frac{2^{n-1}-m(n-1)^{m-1}}{2^n} \leq \frac{1-k_{n+1}^*}{1-k_n^*} \leq \frac{n}{n-m}\frac{2^{n-1}}{2^n-mn^{m-1}}$ (because $\binom{n}{j} \leq n^i$ if $j \leq i$).

Since we have $\lim_{n\to\infty} \frac{n}{n-m}\frac{2^{n-1}-m(n-1)^{m-1}}{2^n} = \frac{1}{2}$, and $\lim_{n\to\infty} \frac{n}{n-m}\frac{2^{n-1}}{2^n-mn^{m-1}} = \frac{1}{2}$, it follows that $\lim_{n\to\infty} \frac{1-k_{n+1}^*}{1-k_n^*} = \frac{1}{2}$. □

2.1.7 Worst-Case Optimality Outside the Family

In this subsection, we prove that the worst-case optimal redistribution mechanism among linear VCG redistribution mechanisms is in fact optimal (in the worst case) among *all* redistribution mechanisms that are deterministic, anonymous, strategy-proof, efficient and satisfy the non-deficit constraint. Thus, restricting our attention to linear VCG redistribution mechanisms did not come at a loss.

To prove this theorem, we need the following lemma. This lemma is not new: it was informally stated by Cavallo [20]. For completeness, we present it here with a detailed proof.

Lemma 2. *A VCG redistribution mechanism is deterministic, anonymous and strategy-proof if and only if there exists a function $f : \mathbf{R^{n-1}} \to \mathbf{R}$, so that the redistribution payment z_i received by i satisfies*

$$z_i = f(v_1, v_2, \ldots, v_{i-1}, v_{i+1}, \ldots, v_n)$$

for all i and all bid vectors.

Proof. First, let us prove the "only if" direction, that is, if a VCG redistribution mechanism is deterministic, anonymous and strategy-proof then there exists a deterministic function $f : \mathbf{R^{n-1}} \rightarrow \mathbf{R}$, which makes $z_i = f(v_1, v_2, \ldots, v_{i-1}, v_{i+1}, \ldots, v_n)$ for all i and all bid vectors.

If a VCG redistribution mechanism is deterministic and anonymous, then for any bid vector $v_1 \geq v_2 \geq \ldots \geq v_n$, the mechanism outputs a unique redistribution payment list: $z_1, z_2, \ldots, z_n$. Let $G : \mathbf{R^n} \rightarrow \mathbf{R^n}$ be the function that maps $v_1, v_2, \ldots, v_n$ to $z_1, z_2, \ldots, z_n$ for all bid vectors. Let $H(i, x_1, x_2, \ldots, x_n)$ be the ith element of $G(x_1, x_2, \ldots, x_n)$, so that $z_i = H(i, v_1, v_2, \ldots, v_n)$ for all bid vectors and all $1 \leq i \leq n$. Because the mechanism is anonymous, two agents should receive the same redistribution payment if their bids are the same. So, if $v_i = v_j$, $H(i, v_1, v_2, \ldots, v_n) = H(j, v_1, v_2, \ldots, v_n)$. Hence, if we let $j = min\{t | v_t = v_i\}$, then $H(i, v_1, v_2, \ldots, v_n) = H(j, v_1, v_2, \ldots, v_n)$.

Let us define $K : \mathbf{R^n} \rightarrow \mathbf{N} \times \mathbf{R^n}$ as follows: $K(y, x_1, x_2, \ldots, x_{n-1}) = [j, w_1, w_2, \ldots, w_n]$, where $w_1, w_2, \ldots, w_n$ are $y, x_1, x_2, \ldots, x_{n-1}$ sorted in descending order, and $j = min\{t | w_t = y\}$. ($\{t | w_t = y\} \neq \emptyset$ because $y \in \{w_1, w_2, \ldots, w_n\}$).

Also let us define $F : \mathbf{R^n} \rightarrow \mathbf{R}$ by $F(v_i, v_1, v_2, \ldots, v_{i-1}, v_{i+1}, \ldots, v_n) = H \circ K(v_i, v_1, v_2, \ldots, v_{i-1}, v_{i+1}, \ldots, v_n) = H(min\{t | v_t = v_i\}, v_1, v_2, \ldots, v_n) = H(i, v_1, v_2, \ldots, v_n) = z_i$. That is, F is the redistribution payment to an agent that bids v_i when the other bids are $v_1, v_2, \ldots, v_{i-1}, v_{i+1}, \ldots, v_n$.

Since our mechanism is required to be strategy-proof, and the space of valuations is unrestricted, z_i should be independent of v_i by Lemma 1 in Cavallo [20]. Hence, we can simply ignore the first variable input to F; let $f(x_1, x_2, \ldots, x_{n-1}) = F(0, x_1, x_2, \ldots, x_{n-1})$. So, we have $z_i = f(v_1, v_2, \ldots, v_{i-1}, v_{i+1}, \ldots, v_n)$ for all bid vectors and i. This completes the proof for the "only if" direction.

For the "if" direction, if the redistribution payment received by i satisfies $z_i = f(v_1, v_2, \ldots, v_{i-1}, v_{i+1}, \ldots, v_n)$ for all bid vectors and i, then this is clearly a deterministic and anonymous mechanism. To prove strategy-proofness, we observe that because an agent's redistribution payment is not affected by her own bid, her incentives are the same as in the VCG mechanism, which is strategy-proof. □

Now we are ready to introduce the next theorem:

Theorem 2. *For any m and n with $n \geq m+2$, the worst-case optimal mechanism among the family of linear VCG redistribution mechanisms is worst-case optimal among all mechanisms that are deterministic, anonymous, strategy-proof, efficient and satisfy the non-deficit constraint.*

While we needed individual rationality earlier, this theorem does not mention it, that is, we cannot find a mechanism with better worst-case performance even if we sacrifice individual rationality. (The worst-case optimal linear VCG redistribution mechanism is of course individually rational.)

Proof. Suppose there is a redistribution mechanism (when the number of units is m and the number of agents is n) that satisfies all of the above properties and has a better worst-case performance than the worst-case optimal linear VCG redistribution mechanism, that is, its worst-case redistribution fraction $\hat{k}$ is strictly greater than k^*.

By Lemma 2, for this mechanism, there is a function $f : \mathbf{R^{n-1}} \to \mathbf{R}$ so that $z_i = f(v_1, v_2, \ldots, v_{i-1}, v_{i+1}, \ldots, v_n)$ for all i and all bid vectors. We first prove that f has the following properties.

Proposition 5. $f(1, 1, \ldots, 1, 0, 0, \ldots, 0) = 0$ *if the number of* 1*s is less than or equal to* m.

Proof of proposition. We assumed that for this mechanism, the worst-case redistribution fraction satisfies $\hat{k} > k^* \geq 0$. If the total VCG payment is x, the total redistribution payment should be in $[\hat{k}x, x]$ (non-deficit criterion). Consider the case where all agents bid 0, so that the total VCG payment is also 0. Hence, the total redistribution payment should be in $[\hat{k} \cdot 0, 0]$—that is, it should be 0. Hence every agent's redistribution payment $f(0, 0, \ldots, 0)$ must be 0.

Now, let $t_i = f(1, 1, \ldots, 1, 0, 0, \ldots, 0)$ where the number of 1s equals i. We proved $t_0 = 0$. If $t_{n-1} = 0$, consider the bid vector where everyone bids 1. The total VCG payment is m and the total redistribution payment is $nf(1, 1, \ldots, 1) = nt_{n-1} = 0$. This corresponds to 0% redistribution, which is contrary to our assumption that $\hat{k} > k^* \geq 0$. Now, consider $j = min\{i|t_i \neq 0\}$ (which is well-defined because $t_{n-1} \neq 0$). If $j > m$, the property is satisfied. If $j \leq m$, consider the bid vector where $v_i = 1$ for $i \leq j$ and $v_i = 0$ for all other i. Under this bid vector, the first j agents each get redistribution payment $t_{j-1} = 0$, and the remaining $n-j$ agents each get t_j. Thus, the total redistribution payment is $(n-j)t_j$. Because the total VCG payment for this bid vector is 0, we must have $(n-j)t_j = 0$. So $t_j = 0$ $(j \leq m < n)$. But this is contrary to the definition of j. Hence $f(1, 1, \ldots, 1, 0, 0, \ldots, 0) = 0$ if the number of 1s is less than or equal to m. □

Proposition 6. *f satisfies the following inequalities:*

$$\hat{k}m \leq (n-m-1)t_{m+1} \leq m$$

$$\hat{k}m \leq (m+i)t_{m+i-1} + (n-m-i)t_{m+i} \leq m \text{ for } i = 2, 3, \ldots, n-m-1$$

$$\hat{k}m \leq nt_{n-1} \leq m$$

Here t_i is defined as in the proof of Proposition 5.

Proof of proposition. For $j = m+1, \ldots, n$, consider the bid vectors where $v_i = 1$ for $i \leq j$ and $v_i = 0$ for all other i. These bid vectors together with the non-deficit constraint and worst-case constraint produce the above set of inequalities: for example, when $j = m+1$, we consider the bid vector $v_i = 1$ for $i \leq m+1$ and $v_i = 0$ for all other i. The first $m+1$ agents each receive a redistribution payment of

$t_m = 0$, and all other agents each receive t_{m+1}. Thus, the total VCG redistribution is $(n-m-1)t_{m+1}$. The non-deficit constraint gives $(n-m-1)t_{m+1} \leq m$ (because the total VCG payment is m). The worst-case constraint gives $(n-m-1)t_{m+1} \geq \hat{k}m$. Combining these two, we get the first inequality. The other inequalities can be obtained in the same way. □

We now observe that the inequalities in Proposition 6, together with $\hat{k} \geq k^*$, are the same as those in Proposition 4 (where the t_i are replaced by the $\hat{x}_i$). Thus, we can conclude that $\hat{k} = k^*$, which is contrary to our assumption $\hat{k} > k^*$. Hence no mechanism satisfying all the listed properties has a redistribution fraction greater than k^* in the worst case. □

So far we have only talked about the case where $n \geq m+2$. For the purpose of completeness, we provide the following proposition for the $n = m+1$ case. (We assume $n > m$ in the unit demand setting.)

Proposition 7. *For any m and n with $n = m+1$, the original VCG mechanism (that is, redistributing nothing) is (uniquely) worst-case optimal among all redistribution mechanisms that are deterministic, anonymous, strategy-proof, efficient and satisfy the non-deficit constraint.*

We recall that when $n = m+1$, Proposition 1 tells us that the only mechanism inside the family of linear redistribution mechanisms is the original VCG mechanism, so that this mechanism is automatically worst-case optimal inside this family. However, to prove the above proposition, we need to show that it is worst-case optimal among *all* redistribution mechanisms that have the desired properties.

Proof. Suppose a redistribution mechanism exists that satisfies all of the above properties and has a worst-case performance as good as the original VCG mechanism, that is, its worst-case redistribution fraction is greater than or equal to 0. This implies that the total redistribution payment of this mechanism is always nonnegative.

By Lemma 2, for this mechanism, there is a function $f : \mathbf{R^{n-1}} \to \mathbf{R}$ so that $z_i = f(v_1, v_2, \ldots, v_{i-1}, v_{i+1}, \ldots, v_n)$ for all i and all bid vectors. We will prove that $f(x_1, x_2, \ldots, x_{n-1}) = 0$ for all $x_1 \geq x_2 \geq \ldots \geq x_{n-1} \geq 0$.

First, consider the bid vector where $v_i = 0$ for all i. Here, each agent receives a redistribution payment $f(0, 0, \ldots, 0)$. The total redistribution payment is then $nf(0, 0, \ldots, 0)$, which should be both greater than or equal to 0 (by the above observation) as well less than or equal to 0 (using the non-deficit criterion and the fact that the total VCG payment is 0). It follows that $f(0, 0, \ldots, 0) = 0$. Now, let us consider the bid vector where $v_1 = x_1 \geq 0$ and $v_i = 0$ for all other i. For this bid vector, the agent with the highest bid receives a redistribution payment of $f(0, 0, \ldots, 0) = 0$, and the other $n - 1$ agents each receive $f(x_1, 0, \ldots, 0)$. By the same reasoning as above, the total redistribution payment should be both greater than or equal to 0 and less than or equal to 0, hence $f(x_1, 0, \ldots, 0) = 0$ for all $x_1 \geq 0$.

Proceeding by induction, let us assume $f(x_1, x_2, \ldots, x_k, 0, \ldots, 0) = 0$ for all $x_1 \geq x_2 \geq \ldots \geq x_k \geq 0$, for some $k < n - 1$. Consider the bid vector where $v_i = x_i$ for $i \leq k + 1$, and $v_i = 0$ for all other i, where the x_i are arbitrary numbers satisfying $x_1 \geq x_2 \geq \ldots \geq x_k \geq x_{k+1} \geq 0$. For the agents with the highest $k + 1$ bids, their redistribution payment is specified by f acting on an input with only k non-zero variables. Hence they all receive 0 by induction assumption. The other $n - k - 1$ agents each receive $f(x_1, x_2, \ldots, x_k, x_{k+1}, 0, \ldots, 0)$. The total redistribution payment is then $(n - k - 1)f(x_1, x_2, \ldots, x_k, x_{k+1}, 0, \ldots, 0)$, which should be both greater than or equal to 0, and less than or equal to the total VCG payment. Now, in this bid vector, the lowest bid is 0 because $k + 1 < n$. But since $n = m + 1$, the total VCG payment is $mv_n = 0$. So we have $f(x_1, x_2, \ldots, x_k, x_{k+1}, 0, \ldots, 0) = 0$ for all $x_1 \geq x_2 \geq \ldots \geq x_k \geq x_{k+1} \geq 0$. By induction, this statement holds for all $k < n-1$; when $k+1 = n-1$, we have $f(x_1, x_2, \ldots, x_{n-2}, x_{n-1}) = 0$ for all $x_1 \geq x_2 \geq \ldots \geq x_{n-2} \geq x_{n-1} \geq 0$. Hence, in this mechanism, the redistribution payment is always 0; that is, the mechanism is just the original VCG mechanism. □

Incidentally, we obtain the following corollary:

Corollary 2. *No VCG redistribution mechanism satisfies all of the following: determinism, anonymity, strategy-proofness, efficiency, and (strong) budget balance. This holds for any $n \geq m+1$.*

Proof. For the case $n \geq m+2$: If such a mechanism exists, its worst-case performance would be better than that of the worst-case optimal linear VCG redistribution mechanism, which by Theorem 1 obtains a redistribution fraction strictly less than 1. But Theorem 2 shows that it is impossible to outperform this mechanism in the worst case.

For the case $n = m+1$: If such a mechanism exists, it would perform as well as the original VCG mechanism in the worst case, which implies that it is identical to the VCG mechanism by Proposition 7. But the VCG mechanism is not (strongly) budget balanced. □

2.1.8 Worst-Case Optimal Mechanism When Deficits Are Allowed

In the previous subsection, we showed that even if the individual rationality requirement is dropped, the worst-case optimal redistribution mechanism remains the same. In this subsection, we consider dropping the non-deficit requirement, and try to find the redistribution mechanism that deviates the least from budget balance (in the worst case).

We define the *imbalance* to be the absolute difference between the total redistribution and the total VCG payment, and define the *imbalance fraction* to be the ratio between the imbalance and the total VCG payment. Our goal is to minimize the worst-case imbalance fraction. Finding the optimal linear mechanism corresponds to the following optimization model:

Minimize k_d (the imbalance fraction in the worst case)
Subject to:
For every bid vector $v_1 \geq v_2 \geq \ldots \geq v_n \geq 0$
$z_n \geq 0$ (individual rationality)
$|z_1 + z_2 + \ldots + z_n - mv_{m+1}| \leq k_d m v_{m+1}$ (imbalance constraint)
We recall that $z_i = c_0 + c_1 v_1 + c_2 v_2 + \ldots + c_{i-1} v_{i-1} + c_i v_{i+1} + \ldots + c_{n-1} v_n$

The imbalance constraint can also be written as

$(1 - k_d) m v_{m+1} \leq z_1 + z_2 + \ldots + z_n \leq (1 + k_d) m v_{m+1}$

The above optimization model can be transformed into a linear program, based on the following observations.

Proposition 8. *If $c_0, c_1, \ldots, c_{n-1}$ satisfy both the individual rationality and the imbalance constraints, then $c_i = 0$ for $i = 0, \ldots, m$.*

The proof is a slight modification of the proof of Proposition 1.

Proof. First, let us prove that $c_0 = 0$. Consider the bid vector in which $v_i = 0$ for all i. To obtain individual rationality, we must have $c_0 \geq 0$. To satisfy the imbalance constraint, we must have $c_0 \leq 0$. Thus we know $c_0 = 0$. Now, if $c_i = 0$ for all i, there is nothing to prove. Otherwise, let $j = min\{i | c_i \neq 0\}$. Assume that $j \leq m$. We recall that we can write the individual rationality constraint as follows: $z_n = c_0 + c_1 v_1 + c_2 v_2 + c_3 v_3 + \ldots + c_{n-2} v_{n-2} + c_{n-1} v_{n-1} \geq 0$ for any bid vector. Let us consider the bid vector in which $v_i = 1$ for $i \leq j$ and $v_i = 0$ for the rest. In this case $z_n = c_j$, so we must have $c_j \geq 0$. The imbalance constraint requires that : $z_1 + z_2 + \ldots + z_n \leq (1 + k_d) m v_{m+1}$ for any bid vector. Consider the same bid vector as above. We have $z_i = 0$ for $i \leq j$, because for these bids, the jth highest other bid has value 0, so all the c_i that are nonzero are multiplied by 0. For $i > j$, we have $z_i = c_j$, because the jth highest other bid has value 1, and all lower bids have value

0. So the imbalance constraint tells us that $c_j(n-j) \leq (1+k_d)mv_{m+1}$. Because $j \leq m$, $v_{m+1} = 0$, so the right hand side is 0. We also have $n - j > 0$ because $j \leq m < n$. So $c_j \leq 0$. Because we have already established that $c_j \geq 0$, it follows that $c_j = 0$; but this is contrary to assumption. So $j > m$. □

Proposition 9. *The imbalance constraint can be written as linear inequalities involving only the* c_i *and* k_d*.*

The proof is a slight modification of the proof of Proposition 3.

Proof. The imbalance constraint requires that for any bid vector, $(1-k_d)mv_{m+1} \leq z_1 + z_2 + \ldots + z_n \leq (1+k_d)mv_{m+1}$, where $z_i = c_0 + c_1v_1 + c_2v_2 + \ldots + c_{i-1}v_{i-1} + c_iv_{i+1} + \ldots + c_{n-1}v_n$ for $i = 1, 2, \ldots, n$. Because $c_i = 0$ for $i \leq m$, we can simplify this inequality to

$$q_{m+1}v_{m+1} + q_{m+2}v_{m+2} + \ldots + q_nv_n \geq 0$$

$$q_{m+1} = (n-m-1)c_{m+1} - (1-k_d)m$$

$$q_i = (i-1)c_{i-1} + (n-i)c_i, \text{ for } i = m+2, \ldots, n-1$$

$$q_n = (n-1)c_{n-1}$$

$$Q_{m+1}v_{m+1} + Q_{m+2}v_{m+2} + \ldots + Q_nv_n \leq 0$$

$$Q_{m+1} = (n-m-1)c_{m+1} - (1+k_d)m$$

$$Q_i = (i-1)c_{i-1} + (n-i)c_i, \text{ for } i = m+2, \ldots, n-1$$

$$Q_n = (n-1)c_{n-1}$$

By Lemma 1, this is equivalent to $\sum_{i=m+1}^{j} q_i \geq 0$ for $j = m+1, \ldots, n$ and $\sum_{i=m+1}^{j} Q_i \leq 0$ for $j = m+1, \ldots, n$. So, we can simplify further as follows:

$$(1-k_d)m \leq (n-m-1)c_{m+1} \leq (1+k_d)m$$

$$(1-k_d)m \leq n\sum_{j=m+1}^{j=m+i-1} c_j + (n-m-i)c_{m+i} \leq (1+k_d)m \text{ for } i = 2, \ldots, n-m-1$$

$(1-k_d)m \leq n\sum_{j=m+1}^{j=n-1} c_j \leq (1+k_d)m$

So, the imbalance constraint can also be written as a set of linear inequalities involving only the c_i and k_d. □

Combining all the propositions (together with Proposition 2), we see that the original optimization problem can be transformed into the following linear program.

Variables: $c_{m+1}, c_{m+2}, \ldots, c_{n-1}, k_d$
Minimize k_d (the imbalance fraction in the worst case)
Subject to:
$\sum_{i=m+1}^{j} c_i \geq 0$ for $j = m+1, \ldots, n-1$
$(1-k_d)m \leq (n-m-1)c_{m+1} \leq (1+k_d)m$
$(1-k_d)m \leq n\sum_{j=m+1}^{j=m+i-1} c_j + (n-m-i)c_{m+i} \leq (1+k_d)m$ for $i = 2, \ldots, n-m-1$
$(1-k_d)m \leq n\sum_{j=m+1}^{j=n-1} c_j \leq (1+k_d)m$

For this model, it is easy to see that k_d is bounded below by 0. Also, $k_d = 1$ and $c_i = 0$ for all i form a feasible solution. So an optimal solution always exists. As in the case where deficits are not allowed, the optimal solution can be analytically characterized. The characterization is the following:

Theorem 3. *For any m and n with $n \geq m+2$, the worst-case optimal mechanism with deficits (among linear VCG redistribution mechanisms) is unique. For this mechanism, the imbalance fraction in the worst case is*

$$k_d^* = \frac{\binom{n-1}{m}}{\sum_{j=m+1}^{n}\binom{n}{j}}$$

The worst-case optimal mechanism with deficits is characterized by the following values for the c_i:

$$c_i^* = \frac{2(-1)^{i+m-1}(n-m)\binom{n-1}{m-1}}{i\sum_{j=m+1}^{n}\binom{n}{j}} \frac{1}{\binom{n-1}{i}} \sum_{j=i}^{n-1}\binom{n-1}{j}$$

for $i = m+1, \ldots, n-1$.

From Proposition 8 it follows that $c_i = 0$ for $i \leq m$.

Proof. Let $\alpha = k_d^*/(1-k^*)$, where k^* is the worst-case optimal redistribution fraction in Theorem 1. To avoid ambiguity, we refer to the c_i^* in Theorem 1 as c_i^{w*}, and to the c_i^* here as c_i^{d*}. Inspection reveals that $c_i^{d*} = 2\alpha c_i^{w*}$ for all i. We have shown in Theorem 1 that

$\sum_{i=m+1}^{j} c_i^{w*} \geq 0$ for $j = m+1, \ldots, n-1$

$k^* m \leq (n-m-1)c_{m+1}^{w*} \leq m$

$k^* m \leq n \sum_{j=m+1}^{j=m+i-1} c_j^{w*} + (n-m-i)c_{m+i}^{w*} \leq m$ for $i = 2, \ldots, n-m-1$

$k^* m \leq n \sum_{j=m+1}^{j=n-1} c_j^{w*} \leq m$

So we have

$\sum_{i=m+1}^{j} c_i^{d*} \geq 0$ for $j = m+1, \ldots, n-1$ (α is positive)

$2\alpha k^* m \leq (n-m-1)c_{m+1}^{d*} \leq 2\alpha m$

$2\alpha k^* m \leq n \sum_{j=m+1}^{j=m+i-1} c_j^{d*} + (n-m-i)c_{m+i}^{d*} \leq 2\alpha m$ for $i = 2, \ldots, n-m-1$

$2\alpha k^* m \leq n \sum_{j=m+1}^{j=n-1} c_j^{d*} \leq 2\alpha m$

A sequence of algebraic manipulations reveals that $2\alpha k^* = (1-k_d^*)$ and $2\alpha = (1+k_d^*)$. Hence, k_d^* and the c_i^{d*} form a feasible solution, because we have

$\sum_{i=m+1}^{j} c_i^{d*} \geq 0$ for $j = m+1, \ldots, n-1$

$(1-k_d^*)m \leq (n-m-1)c_{m+1}^{d*} \leq (1+k_d^*)m$

$(1-k_d^*)m \leq n \sum_{j=m+1}^{j=m+i-1} c_j^{d*} + (n-m-i)c_{m+i}^{d*} \leq (1+k_d^*)m$ for $i = 2, \ldots, n-m-1$

$(1-k_d^*)m \leq n \sum_{j=m+1}^{j=n-1} c_j^{d*} \leq (1+k_d^*)m$

We proceed to show that it is in fact the unique optimal solution. Suppose $\hat{c}_i$ and $\hat{k}_d$ form a feasible solution, and $\hat{k}_d \leq k_d^*$. We have

$(1-k_d^*)m \leq (1-\hat{k}_d)m \leq (n-m-1)\hat{c}_{m+1} \leq (1+\hat{k}_d)m \leq (1+k_d^*)m$

$(1-k_d^*)m \leq (1-\hat{k}_d)m \leq n\sum_{j=m+1}^{j=m+i-1}\hat{c}_j + (n-m-i)\hat{c}_{m+i} \leq (1+\hat{k}_d)m \leq (1+k_d^*)m$

for $i = 2, \ldots, n-m-1$

$(1-k_d^*)m \leq (1-\hat{k}_d)m \leq n\sum_{j=m+1}^{j=n-1}\hat{c}_j \leq (1+\hat{k}_d)m \leq (1+k_d^*)m$

We introduce new variables $x_{m+1}, x_{m+2}, \ldots, x_{n-1}$, defined by $x_j = \frac{1}{2\alpha}\sum_{i=m+1}^{j}\hat{c}_i$ for $j = m+1, \ldots, n-1$. The above inequalities can be rewritten in terms of x_i, we have

$k^*m \leq (n-m-1)x_{m+1} \leq m$

$k^*m \leq (m+i)x_{m+i-1} + (n-m-i)x_{m+i} \leq m$ for $i = 2, \ldots, n-m-1$

$k^*m \leq nx_{n-1} \leq m$

However, in Proposition 4, we proved that these inequalities have a unique solution. Therefore, there is only one value that each of $\hat{c}_i$ and $\hat{k}_d$ can have. This proves that k_d^* and the c_i^{d*} form the unique optimal solution. □

$\alpha = k_d^*/(1-k^*)$ can be interpreted as the ratio between the imbalance fraction of the worst-case optimal mechanism with deficits (among linear VCG redistribution mechanisms) and the imbalance fraction of the worst-case optimal mechanism without deficits. This ratio can be expressed as follows:

$\alpha = k_d^*/(1-k^*) = \frac{\sum_{j=m}^{n-1}\binom{n-1}{j}}{\sum_{j=m+1}^{n}\binom{n}{j}} = \frac{\sum_{j=m+1}^{n}\binom{n-1}{j-1}}{\sum_{j=m+1}^{n}\binom{n}{j}} = \frac{\sum_{j=m+1}^{n}((j/n)\binom{n}{j})}{\sum_{j=m+1}^{n}\binom{n}{j}}$

For fixed n, this ratio increases as m increases. (This is because as we decrease m by 1, the ratio of the additional terms in the fraction decreases.) When $m = 1$, $\alpha = \frac{2^{n-1}-1}{2^n-n-1}$ (for large n, roughly $\frac{1}{2}$); when $m = n-2$, $\alpha = \frac{n}{n+1}$ (for large n, roughly 1). Hence, if m is small (relative to n), the worst-case optimal linear VCG redistribution mechanism with deficits is much closer to budget balance than the worst-case optimal mechanism without deficits; if m is large (relative to n), they are

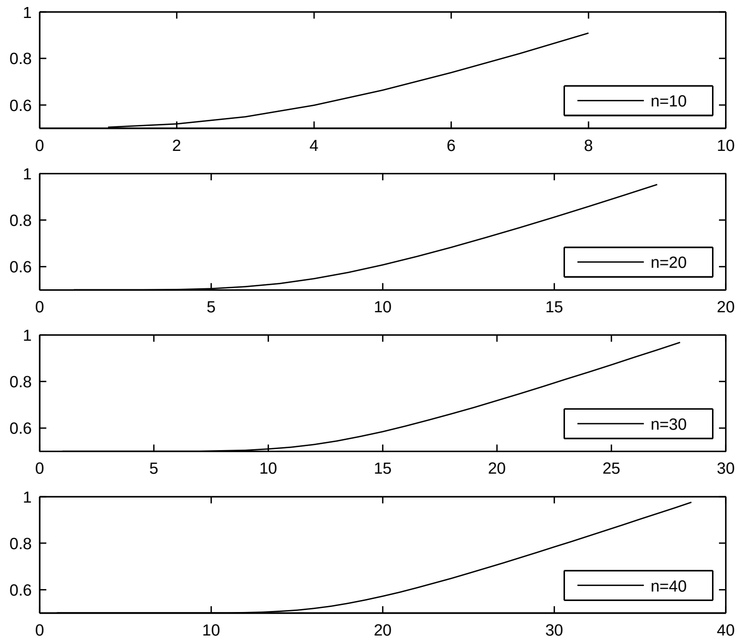

FIGURE 2.2: The ratio between the imbalance fractions of the worst-case optimal mechanisms with and without deficits.

about the same. On the other hand, when m is small relative to n, then the worst-case optimal redistribution fraction is large even with the non-deficit requirement. This means that the non-deficit constraint does not come at a great cost. Figure 2.2 shows how α changes as a function of m and n.

Now we prove that the worst-case optimal linear VCG redistribution mechanism with deficits is in fact optimal among *all* redistribution mechanisms that are deterministic, anonymous, strategy-proof and efficient.

Theorem 4. *For any m and n with $n \geq m + 2$, the worst-case optimal mechanism with deficits among linear VCG redistribution mechanisms has the smallest worst-case imbalance fraction among all VCG redistribution mechanisms that are deterministic, anonymous, strategy-proof and efficient.*

As in the case of Theorem 2, there is no redistribution mechanism with a smaller worst-case imbalance fraction even if we sacrifice individual rationality.

Proof. Suppose there is a redistribution mechanism (when the number of units is m and the number of agents is n) that satisfies all of the above properties and has a smaller worst-case imbalance fraction than that of the worst-case optimal linear VCG redistribution mechanism with deficits—that is, its worst-case imbalance fraction $\hat{k}_d$ is strictly less than k_d^*.

By Lemma 2, for this mechanism, there is a function $f : \mathbf{R^{n-1}} \to \mathbf{R}$ so that $z_i = f(v_1, v_2, \ldots, v_{i-1}, v_{i+1}, \ldots, v_n)$ for all i and all bid vectors. The following properties of f follow from straightforward modifications of the proofs of Proposition 5 and Proposition 6.

Proposition 10. *$f(1, 1, \ldots, 1, 0, 0, \ldots, 0) = 0$ if the number of $1s$ is less than or equal to m.*

Proposition 11. *f satisfies the following inequalities:*

$$(1 - \hat{k}_d)m \leq (n - m - 1)t_{m+1} \leq (1 + \hat{k}_d)m$$

$$(1 - \hat{k}_d)m \leq (m + i)t_{m+i-1} + (n - m - i)t_{m+i} \leq (1 + \hat{k}_d)m \text{ for } i = 2, 3, \ldots, n - m - 1$$

$$(1 - \hat{k}_d)m \leq nt_{n-1} \leq (1 + \hat{k}_d)m$$

$t_i = f(1, 1, \ldots, 1, 0, 0, \ldots, 0)$ where the number of $1s$ equals i

Let $x_i = \frac{1}{2\alpha}t_i$ for $i = m + 1, \ldots, n - 1$. Since $\hat{k}_d < k_d^*$, we have

$k^*m < \frac{1}{2\alpha}(1 - \hat{k}_d)m \leq (n - m - 1)x_{m+1} \leq \frac{1}{2\alpha}(1 + \hat{k}_d)m < m$

$k^*m < \frac{1}{2\alpha}(1 - \hat{k}_d)m \leq (m + i)x_{m+i-1} + (n - m - i)x_{m+i} \leq \frac{1}{2\alpha}(1 + \hat{k}_d)m < m$ for $i = 2, 3, \ldots, n - m - 1$

$k^*m < \frac{1}{2\alpha}(1 - \hat{k}_d)m \leq nx_{n-1} \leq \frac{1}{2\alpha}(1 + \hat{k}_d)m < m$

By Proposition 4, the above system of inequalities cannot hold. Hence no mechanism satisfying all the listed properties has an imbalance fraction less than k_d^* in the worst case. □

For the purpose of completeness, we note the following proposition, which follows from a straightforward modification of the proof of Proposition 7.

Proposition 12. *For any m and n with $n = m + 1$, the original VCG mechanism (that is, redistributing nothing) is (uniquely) the worst-case optimal mechanism with deficits among all redistribution mechanisms that are deterministic, anonymous, strategy-proof and efficient.*

Proof. Suppose a redistribution mechanism exists that satisfies all of the above properties and has a worst-case performance as good as the original VCG mechanism, that is, its worst-case imbalance fraction is less than or equal to 100%.

By Lemma 2, for this mechanism, there is a function $f : \mathbf{R^{n-1}} \to \mathbf{R}$ so that $z_i = f(v_1, v_2, \ldots, v_{i-1}, v_{i+1}, \ldots, v_n)$ for all i and all bid vectors. We will prove that $f(x_1, x_2, \ldots, x_{n-1}) = 0$ for all $x_1 \geq x_2 \geq \ldots \geq x_{n-1} \geq 0$.

First, consider the bid vector where $v_i = 0$ for all i. Here, each agent receives a redistribution payment $f(0, 0, \ldots, 0)$. The total redistribution payment is then $nf(0, 0, \ldots, 0)$, which should be 0, because the total VCG payment is 0 (under 100% imbalance fraction, the imbalance is still 0). It follows that $f(0, 0, \ldots, 0) = 0$. Now, let us consider the bid vector where $v_1 = x_1 \geq 0$ and $v_i = 0$ for all other i. For this bid vector, the agent with the highest bid receives a redistribution payment of $f(0, 0, \ldots, 0) = 0$, and the other $n - 1$ agents each receive $f(x_1, 0, \ldots, 0)$. By the same reasoning as above, the total redistribution payment should be 0, hence $f(x_1, 0, \ldots, 0) = 0$ for all $x_1 \geq 0$.

Proceeding by induction, let us assume $f(x_1, x_2, \ldots, x_k, 0, \ldots, 0) = 0$ for all $x_1 \geq x_2 \geq \ldots \geq x_k \geq 0$, for some $k < n - 1$. Consider the bid vector where $v_i = x_i$ for $i \leq k + 1$, and $v_i = 0$ for all other i, where the x_i are arbitrary numbers satisfying $x_1 \geq x_2 \geq \ldots \geq x_k \geq x_{k+1} \geq 0$. For the agents with the highest $k + 1$ bids, their redistribution payment is specified by f acting on an input with only k

non-zero variables. Hence they all receive 0 by induction assumption. The other $n-k-1$ agents each receive $f(x_1, x_2, \dots, x_k, x_{k+1}, 0, \dots, 0)$. The total redistribution payment is then $(n-k-1)f(x_1, x_2, \dots, x_k, x_{k+1}, 0, \dots, 0)$. Now, in this bid vector, the lowest bid is 0 because $k+1 < n$. But since $n = m+1$, the total VCG payment is $mv_n = 0$, which forces the total redistribution payment to be 0. So we have $f(x_1, x_2, \dots, x_k, x_{k+1}, 0, \dots, 0) = 0$ for all $x_1 \geq x_2 \geq \dots \geq x_k \geq x_{k+1} \geq 0$. By induction, this statement holds for all $k < n-1$; when $k+1 = n-1$, we have $f(x_1, x_2, \dots, x_{n-2}, x_{n-1}) = 0$ for all $x_1 \geq x_2 \geq \dots \geq x_{n-2} \geq x_{n-1} \geq 0$. Hence, in this mechanism, the redistribution payment is always 0; that is, the mechanism is just the original VCG mechanism. □

2.1.9 Multi-Unit Auction with Nonincreasing Marginal Values

In this subsection, we consider the more general setting where the agents have nonincreasing marginal values. (Units remain indistinguishable.) An agent's bid is now a vector of m elements, with the jth element denoting this agent's marginal value for getting her jth unit (and the elements are nonincreasing in j). That is, the agent's valuation for receiving j units is the sum of the first j elements. Let the set of agents be $\{1, 2, \dots, n\}$, where i is the agent with the ith highest initial marginal value (the marginal value for winning the first unit).

We still consider only the case where $m \leq n-2$, because if $m \geq n-1$, then the original VCG mechanism is worst-case optimal, both with and without deficits (we will show this in Proposition 18).

The VCG mechanism requires us to find the efficient allocation. Because marginal values are nonincreasing, this can be achieved by the following greedy algorithm. At each step, we sort the agents according to their upcoming marginal values (their values for winning their next unit), and allocate one unit to the agent with the highest such value. We continue until there are no units left, or the remaining agents all have upcoming marginal values of zero (in this case, we simply throw away the remaining units). Given that marginal values are nonincreasing, the following greedy algorithm is effectively the same (in terms of the allocation process): sort *all* the marginal values (not just those for upcoming units), and accept them in decreasing order. Because marginal values are nonincreasing, when we accept one of them, this marginal value does in fact correspond to that agent's utility for receiving another unit at that point. In the proofs below, this greedy algorithm will provide a useful view of how units are allocated.

In the efficient allocation, only agents $1, \ldots, m$ can possibly win, and the VCG payments are determined by the bids of $1, \ldots, m+1$ (because when we remove an agent, only the top m remaining agents can possibly win).

We will generalize the worst-case optimal mechanism (both with and without deficits) to the current setting, and show in each case that the generalized mechanism has the same worst-case performance. This implies that there does not exist another redistribution mechanism with better worst-case performance (because such a mechanism would also have better worst-case performance in the more specific unit demand setting).

We still use I to denote the set of all agents. We use $-i$ to denote the set of agents other than i. Because the mechanisms under consideration are strategy-proof, agents can be expected to report truthfully; hence, we do not make a sharp distinction between an agent and her bid. We define the following functions:

- $VCG : \mathcal{P}(I) \to \mathbf{R}$

  For any subset S of I, let $VCG(S)$ be the total VCG payment when only the agents in S participate in the auction.

- $E : \mathcal{P}(I) \to \mathbf{R}$

For any subset S of I, let $E(S)$ be the total efficiency (that is, the total utility not taking payments into account) when only the agents in S participate in the auction.

- $e : \mathcal{P}(I) \times I \to \mathbf{R}$

  For any subset S of I and any $a \in S$, let $e(S, a)$ be the utility (not taking payments into account) of agent a, when only the agents in S participate in the auction. We note that $E(S) = \sum_{a \in S} e(S, a)$.

- $U : \mathcal{P}(I) \times \mathbf{N} \to \mathcal{P}(I)$

  For any subset S of I, any integer i ($1 \leq i \leq |S|$), let $U(S, i)$ be the set that results after removing the agent with the ith highest initial marginal value in S from S. (If there is a tie, this tie is broken according to the original order $1, \ldots, n$.)

- $R : \mathcal{P}(I) \times \mathbf{N} \to \mathbf{R}$

  For any subset S of I, any integer i ($0 \leq i \leq |S| - m$), let $R(S, i) = \frac{1}{m+i} \sum_{j=1}^{m+i} R(U(S, j), i - 1)$ if $i > 0$, and $R(S, 0) = VCG(S)$. We emphasize that this is a recursive definition: for $i > 0$, $R(S, i)$ is obtained by computing, for each j with $1 \leq j \leq m + i$, $R(U(S, j), i - 1)$ (that is, the value of the function R after removing the jth agent in S from S, and decreasing i by one), and taking the average. For $i = 0$, it is simply the total VCG payment if only the agents from S are present. Shortly, we will prove some properties of this function that clarify its usefulness to our mechanism.

Let $V_i = R(I, i)$ for all i ($0 \leq i \leq n - m$). In what follows, we present several propositions.

Proposition 13. *For any $S, \hat{S} \in \mathcal{P}(I)$, if $S \subseteq \hat{S}$, then $VCG(S) \leq VCG(\hat{S})$. That is, revenue is nondecreasing in agents.*

The above proposition was proved in [7]. It should be noted that revenue may not be nondecreasing in agents in more general settings [96, 7, 35, 108, 110, 111].

Proposition 14. *For any $S \in \mathcal{P}(I)$, $0 \leq i \leq |S| - m - 2$, and $m + i + 2 \leq j \leq |S|$, we have $R(S, i) = R(U(S, j), i)$.*

Proof. We prove this proposition by induction on i. For $i = 0$ and $j \geq m + 2$, we have $R(S, i) = VCG(S) = VCG(U(S, j)) = R(U(S, j), i)$, because, as we noted earlier, the total VCG payment depends only on the agents with the highest $m + 1$ initial marginal values in S, so removing the jth agent does not change the total VCG payment. Let us assume that we have proven that for $i = k$, if $j \geq m + k + 2$, $R(S, k) = R(U(S, j), k)$. Now let us consider the case where $i = k + 1$. By definition, $R(S, k + 1) = \frac{1}{m+k+1} \sum_{l=1}^{m+k+1} R(U(S, l), k)$. When $j \geq m + i + 2 = m + k + 3$, we can use the induction assumption (using the fact that $j - 1 \geq m + k + 2$) to show that $R(U(S, l), k) = R(U(U(S, l), j - 1), k)$. Hence, $R(S, k+1) = \frac{1}{m+k+1} \sum_{l=1}^{m+k+1} R(U(S, l), k) = \frac{1}{m+k+1} \sum_{l=1}^{m+k+1} R(U(U(S, l), j-1), k) = \frac{1}{m+k+1} \sum_{l=1}^{m+k+1} R(U(U(S, j), l), k) = R(U(S, j), k + 1)$. (In the second-to-last step, the same agents are removed in a different order, although the agents' indices change as other agents are removed.) Hence the proposition is also true for $i = k + 1$. □

Proposition 15. *For any $S, \hat{S} \in \mathcal{P}(I)$, $0 \leq i \leq |S| - m$, if $S \subseteq \hat{S}$, then $R(S, i) \leq R(\hat{S}, i)$. That is, R is nondecreasing in agents.*

Proof. We prove this proposition by induction on i. When $i = 0$, using Proposition 13, $R(S, i) = VCG(S) \leq VCG(\hat{S}) = R(\hat{S}, i)$. Let us assume that we have proven that the proposition is true for $i = k$, that is, $R(S, k) \leq R(\hat{S}, k)$ if $S \subseteq \hat{S}$. Now let us consider the case where $i = k+1$. If $\hat{S}$ and S are the same, the proposition is trivial. Now suppose that $\hat{S}$ has one more agent than S, and that this additional agent has the qth highest initial marginal value in $\hat{S}$. If $q \geq m + k + 2$, $U(S, j) \subseteq U(\hat{S}, j)$

for all $j \leq m+k+1$. By the induction assumption, we have $R(\hat{S}, k+1) = \frac{1}{m+k+1}\sum_{j=1}^{m+k+1} R(U(\hat{S}, j), k) \geq \frac{1}{m+k+1}\sum_{j=1}^{m+k+1} R(U(S, j), k) = R(S, k+1)$.

If $q \leq m+k+1$, $U(S, j) \subseteq U(\hat{S}, j)$ for $j \leq q-1$, and $U(S, j-1) \subseteq U(\hat{S}, j)$ for $q+1 \leq j \leq m+k+1$. Using the induction assumption, we have $R(\hat{S}, k+1) = \frac{1}{m+k+1}\sum_{j=1}^{m+k+1} R(U(\hat{S}, j), k) = \frac{1}{m+k+1}\sum_{j=1}^{q-1} R(U(\hat{S}, j), k) + \frac{1}{m+k+1}\sum_{j=q+1}^{m+k+1} R(U(\hat{S}, j), k) + \frac{1}{m+k+1} R(U(\hat{S}, q), k) \geq \frac{1}{m+k+1}\sum_{j=1}^{q-1} R(U(S, j), k) + \frac{1}{m+k+1}\sum_{j=q+1}^{m+k+1} R(U(S, j-1), k) + \frac{1}{m+k+1} R(S, k) \geq \frac{1}{m+k+1}\sum_{j=1}^{m+k} R(U(S, j), k) + \frac{1}{m+k+1} R(U(S, m+k+1), k) = R(S, k+1)$.

So, if $\hat{S}$ has one more element than S, then $R(S, k+1) \leq R(\hat{S}, k+1)$. It naturally follows that if $\hat{S}$ has even more elements, then we still have $R(S, k+1) \leq R(\hat{S}, k+1)$. □

Proposition 16. *For any $S \in \mathcal{P}(I)$, $R(S, i)$ is nonincreasing in i. In particular, setting $S = I$, V_i is nonincreasing in i.*

Proof. Using Proposition 15, $R(S, i+1) = \frac{1}{m+i+1}\sum_{j=1}^{m+i+1} R(U(S, j), i) \leq \frac{1}{m+i+1}\sum_{j=1}^{m+i+1} R(S, i) = R(S, i)$. □

Proposition 17. *For $0 \leq i \leq n-m-1$, $\sum_{j=1}^{n} R(-j, i) = (n-m-1-i)V_i + (m+1+i)V_{i+1}$.*

Proof. Using Proposition 14, we have $\sum_{j=1}^{n} R(-j, i) = \sum_{j=1}^{m+i+1} R(-j, i) + \sum_{j=m+i+2}^{n} R(-j, i) = (m+i+1)R(I, i+1) + (n-m-i-1)R(I, i) = (m+i+1)V_{i+1} + (n-m-i-1)V_i$. □

Now that we have established these basic properties of R, we are ready to introduce the generalization of the worst-case optimal redistribution mechanism (both with or without deficits) to the setting where agents have nonincreasing marginal values over units.

Theorem 5. *When agents have nonincreasing marginal values over units, for any m and n with $n \geq m+2$, the worst-case optimal redistribution fraction (without deficits) is*

$$k^* = 1 - \frac{\binom{n-1}{m}}{\sum_{j=m}^{n-1} \binom{n-1}{j}}$$

(the same as in Theorem 1), and the worst-case imbalance fraction (with deficits) is

$$k_d^* = \frac{\binom{n-1}{m}}{\sum_{j=m+1}^{n} \binom{n}{j}}$$

(the same as in Theorem 3).

In each case, the following is a worst-case optimal mechanism: to agent i, redistribute $\frac{1}{m}\sum_{j=m+1}^{n-1} c_j^* R(-i, j-m-1)$. Here, the c_j^* from Theorem 1 are used to maximize the worst-case redistribution fraction without deficits, and the c_j^* from Theorem 3 are used to minimize the worst-case imbalance fraction when deficits are allowed. The mechanisms obtained in this way in fact generalize the mechanisms from Theorem 1 and Theorem 3.

Proof. In each case, the mechanism is strategy-proof because each agent's redistribution payment is independent of her own bid ($-i$ does not contain i). It is deterministic, efficient and anonymous. Because $R(-i, j-m-1)$ is nonincreasing in j, and $\sum_{j=m+1}^{i} c_j^* \geq 0$ for $i = m+1, \ldots, n-1$, it follows by Lemma 1 that the mechanism is also individually rational.

Now, we recall that in the unit demand setting, for any bid vector $v_1 \geq v_2 \geq \ldots \geq v_n$, the total amount redistributed by the worst-case optimal mechanism is $\sum_{j=m+1}^{n-1} c_j^*((n-j)v_j + jv_{j+1})$, which is always at least $k^* m v_{m+1}$ and at most $m v_{m+1}$ when we use the c_j^* from Theorem 1; and which is always at least $(1-k_d^*) m v_{m+1}$ and at most $(1+k_d^*) m v_{m+1}$ when we use the c_j^* from Theorem 3. We next show that analogous bounds apply to the more general mechanisms, which will complete the proof.

For the more general mechanisms, the total redistribution payment is $\frac{1}{m}\sum_{i=1}^{n}\sum_{j=m+1}^{n-1} c_j^* R(-i, j-m-1) = \frac{1}{m}\sum_{j=m+1}^{n-1} c_j^* \sum_{i=1}^{n} R(-i, j-m-1)$ $= \frac{1}{m}\sum_{j=m+1}^{n-1} c_j^*((n-j)V_{j-m-1} + jV_{j-m})$. This expression is very similar to the total redistributed by the mechanisms in the unit demand setting: the only differences are that each v_j has been replaced by the V_{j-m-1}, and there is an additional factor $\frac{1}{m}$. Now, the bounds for the unit demand setting hold for *any* nonincreasing sequence of v_j; and, by Proposition 16, we have $V_0 \geq V_1 \geq \ldots \geq V_{n-m-1}$. Hence, $\frac{1}{m}\sum_{j=m+1}^{n-1} c_j^*((n-j)V_{j-m-1} + jV_{j-m})$ is in $[k^* V_0, V_0]$ when we use the c_j^* from Theorem 1, and in $[(1-k_d^*)V_0, (1+k_d^*)V_0]$ when we use the c_j^* from Theorem 3. Because $V_0 = R(I, 0) = VCG(I)$ is the total VCG payment, this proves the result. □

So far we have only talked about the case where $n \geq m+2$. For the purpose of completeness, we provide the following proposition for the $n \leq m+1$ case.

Proposition 18. *For any m and n with $n \leq m+1$, the original VCG mechanism (that is, redistributing nothing) is worst-case optimal, both with or without deficits, among all redistribution mechanisms that are deterministic, anonymous, strategy-proof and efficient.*

The proof of this proposition is based on the "nullification" technique proposed in Cavallo [20].

Proof. Suppose there is a mechanism that satisfies all the desirable properties and has a worst-case performance that is at least as good as the VCG mechanism. Because the mechanism is strategy-proof, the redistribution payment received by an agent should be independent of her own bid. Also, if a bid profile results in a total VCG payment of 0, then under this profile, the total redistribution payment must also be 0.

(If the objective is to maximize redistribution without deficits, negative total redistribution would result in worse performance than VCG, and positive redistribution would violate the non-deficit constraint. If the objective is to minimize imbalance, either negative or positive redistribution would result in worse performance than VCG. These arguments are analogous to those in the proofs of Proposition 7 and Proposition 12.)

For the purpose of this proof only, we introduce the following notation. If an agent has marginal value 1 for every unit among the first k units, and 0 for any further units, we denote her bid by k. These are the only bids that we will use in this proof. For $b_i \in \mathbf{N}$, let $f(b_1, b_2, \ldots, b_{n-1})$ be the redistribution payment received by an agent if the other agents' bids are $b_1, \ldots, b_{n-1}$.

We will prove that for any set of nonnegative integers $b_1, b_2, \ldots, b_{n-1}$, if $\sum_{i=1}^{n-1} b_i \leq m$, we have $f(b_1, \ldots, b_{n-1}) = 0$. We will do so by proving by induction on k ($k \leq m$) the proposition that for any set of nonnegative integers $b_1, b_2, \ldots, b_{n-1}$, if $\sum_{i=1}^{n-1} b_i \leq k$, we have $f(b_1, \ldots, b_{n-1}) = 0$.

For the case $k = 0$, let us consider the case where all the agents bid 0, so that the total redistribution payment is $nf(0, 0, \ldots, 0)$. Because the total VCG payment is 0, the total redistribution must be 0, therefore $f(0, 0, \ldots, 0)$ must be 0.

Now let us assume that for any set of nonnegative integers $b_1, b_2, \ldots, b_{n-1}$, if $\sum_{i=1}^{n-1} b_i \leq k$, we have $f(b_1, \ldots, b_{n-1}) = 0$. Let $b'_1, b'_2, \ldots, b'_{n-1}$ be any set of nonnegative integers that satisfies $\sum_{i=1}^{n-1} b'_i = k + 1$. Consider the bid profile (consisting of n bids) formed by the b'_i and one 0. The redistribution payment received by the agent that bids 0 is then $f(b'_1, b'_2, \ldots, b'_{n-1})$. We note that some of the b'_i may equal 0 as well; by anonymity, the payment for these agents should be the same. The redistribution payment received by any agent that does not bid 0 is 0 by the induction assumption. Hence, the total redistribution is a positive multiple of $f(b'_1, b'_2, \ldots, b'_{n-1})$. Given that

$k+1 \leq m$, the total VCG payment is 0, so it must be that $f(b'_1, b'_2, \ldots, b'_{n-1}) = 0$, completing the proof by induction.

Having proved this, we now find an example with positive total VCG payment but zero total redistribution, which will complete the proof. We recall $m \geq n - 1$. Let us consider the bid profile where one agent bids $m-n+2$ and the other agents each bid 1. Then, the total redistribution payment is $(n-1)f(m-n+2, \underbrace{1,\ldots,1}_{n-2}) + f(\underbrace{1,\ldots,1}_{n-1}) = 0$ (since the previous proposition applies to both $f(m-n+2, \underbrace{1,\ldots,1}_{n-2})$ and $f(\underbrace{1,\ldots,1}_{n-1})$). However, the total VCG payment is positive. Hence, the mechanism has a redistribution fraction of 0% and an imbalance fraction of 100% on this instance. □

2.1.10 General Multi-Unit Auctions

In Subsection 2.1.9, we showed how the results for the unit demand setting can be generalized to the setting where agents have nonincreasing marginal values over the units. The natural next question is whether they can be generalized even further. In this subsection, we study multi-unit settings without any constraint on the bidders' valuations—that is, marginal values can be increasing (but they cannot be negative: units can always be freely disposed of). We show that when there are at least two units, the original VCG mechanism (that is, redistributing nothing) is worst-case optimal, both with and without deficits. (When there is only a single unit, then the agents must have unit demand, so the previous results do apply.)

Proposition 19. *In multi-unit auctions without any restrictions on agents' valuations, when the number of units m is at least 2, the original VCG mechanism (that is, redistributing nothing) is worst-case optimal, both with or without deficits, among all redistribution mechanisms that are deterministic, anonymous, individually rational,* *strategy-proof and efficient.*

The proof of this proposition is based on the "nullification" technique proposed in Cavallo [20]. We emphasize that unlike some of the earlier proofs, this proof does require individual rationality.

Proof. Proposition 18 already established that for $n - 2 < m$, the original VCG mechanism is worst-case optimal even when we do assume nonincreasing marginal values, so it suffices to consider only the case where $n - 2 \geq m$. Suppose there is a mechanism that satisfies all the desirable properties and has a worst-case performance that is at least as good as the original VCG mechanism. Because the mechanism is strategy-proof, the redistribution payment received by an agent should be independent of her own bid.

Also, if a bid profile results in a total VCG payment of 0, then under this profile, the total redistribution payment must also be 0 (otherwise, the performance is worse than that of the original VCG mechanism).

For the purpose of this proof only, we introduce the following notations. If an agent has marginal value 0 for every unit among the first $m - 1$ units, and marginal value 1 for the mth unit, we denote her bid by B_1. If an agent has marginal value 1 for the first unit, and 0 for any further units, we denote her bid by B_2. If an agent has marginal value 0 for all units, we denote her bid by 0. These are the only bids that we will use in this proof. For $b_i \in \{B_1, B_2, 0\}$, let $f(b_1, b_2, \ldots, b_{n-1})$ be the redistribution payment received by an agent if the other agents' bids are $b_1, \ldots, b_{n-1}$. We need $f(b_1, b_2, \ldots, b_{n-1}) \geq 0$ to ensure individual rationality.

We will prove the following:

- $f(0, 0, \ldots, 0) = 0$
- $f(B_1, 0, \ldots, 0) = 0$

- $f(B_2, 0, \ldots, 0) = 0$
- $f(B_1, B_2, 0, \ldots, 0) = 0$

For $f(0, 0, \ldots, 0)$, let us consider the case where all the agents bid 0, so that the total redistribution payment is $nf(0, 0, \ldots, 0)$. Because the total VCG payment is 0, the total redistribution must be 0, therefore $f(0, 0, \ldots, 0)$ must be 0.

For $f(B_1, 0, \ldots, 0)$, let us consider the case where one agent bids B_1 and all the other agents bid 0, so that the total redistribution payment is $(n-1)f(B_1, 0, \ldots, 0) + f(0, 0, \ldots, 0) = (n-1)f(B_1, 0, \ldots, 0)$. Because the total VCG payment is 0, the total redistribution must be 0, therefore $f(B_1, 0, \ldots, 0)$ must be 0. The same argument can be used to show that $f(B_2, 0, \ldots, 0) = 0$.

For $f(B_1, B_2, 0, \ldots, 0)$, let us consider the case where one agent bids B_1, two agents bid B_2 and all the other agents bid 0, so that the total redistribution payment is $(n-3)f(B_1, B_2, B_2, 0, \ldots, 0) + 2f(B_1, B_2, 0, \ldots, 0) + f(B_2, B_2, 0, \ldots, 0)$. However, the total VCG payment is still 0 for these bids (the agents that bid B_2 win; if one of them is removed, we can do no better than to still allocate one unit to the other B_2 agent, and nothing to the other agents—hence each B_2 agent pays 0). Hence, the total redistribution must be 0. Because f is nonnegative everywhere, it follows that $f(B_1, B_2, 0, \ldots, 0)$ must equal 0.

Having proved this, we now find an example with positive total VCG payment but zero total redistribution, which will complete the proof. Let us consider the bid profile where one agent bids B_1, one agent bids B_2, and the other agents all bid 0. Then, the total redistribution payment is $(n-2)f(B_1, B_2, 0, \ldots, 0) + f(B_1, 0, \ldots, 0) + f(B_2, 0, \ldots, 0) = 0$. However, the total VCG payment is positive (because we can accept at most one of the B_1 bid and the B_2 bid). Hence, the mechanism has a redistribution fraction of 0% and an imbalance fraction of 100% on this instance. □

2.2 Optimal-in-Expectation Redistribution Mechanisms

So far, we have evaluated how well a redistribution mechanism does by focusing on the worst case. This is a very robust criterion, but if we have a prior distribution over the valuations, it may make more sense to maximize the *expected* redistribution. In this section, we study the problem of designing VCG redistribution mechanisms that redistribute the most in expectation. From Subsection 2.2.1 to Subsection 2.2.4, we focus on multi-unit auctions with unit demand. In Subsection 2.2.1, we cover the necessary background and introduce our notation. In Subsection 2.2.2, we recall the definition of linear redistribution mechanisms and we solve for optimal-in-expectation linear (OEL) redistribution mechanisms in our setting. We focus on deriving an analytical characterization of these OEL mechanisms. In Subsection 2.2.3, we show how to automatically (using linear programming) solve for (possibly nonlinear) mechanisms that are close to optimal, based on a discretization of the valuation space. This technique is only effective for cases with small number of agents. That is, it does not scale very well. Fortunately, the experimental results in Subsection 2.2.4 show that for auctions with many bidders, the optimal linear mechanism redistributes almost everything, whereas for auctions with few bidders, we can solve for the optimal discretized redistribution mechanism with a very small step size. That is, the two approaches are in some sense complementary. Finally, in Subsection 2.2.5, we study the more general setting of multi-unit auctions with nonincreasing marginal values. We extend the notion of linear redistribution mechanisms to this more general setting, and propose several models for finding optimal linear redistribution mechanisms. It is more difficult to work in this more general setting, since we also need to consider a type of ordering information; we discuss these difficulties in that subsection.

2.2.1 Formalization

From this subsection to Subsection 2.2.4, we focus on multi-unit auctions with unit demand. We still use n and m to denote the number of agents and the number of units. Since we are dealing with the unit demand setting, we assume $n > m$.[8] (Otherwise, it is clearly optimal to give every agent a unit and charge nothing.) As usual, for the ith agent, we denote her true/reported type/bid for winning one unit by v_i (we are focusing on strategy-proof mechanisms). Without losing generality, we assume that $v_1 \geq v_2 \geq \ldots \geq v_n \geq 0$.

Let constants L and U be the lower and upper bounds, respectively, on the possible values. Hence, $\infty > U \geq v_1 \geq v_2 \geq \ldots \geq v_n \geq L \geq 0$. We assume that we have a prior joint probability distribution over the agents' values v_i. We denote the probability density function of this joint distribution by $f(v_1, \ldots, v_n)$. We emphasize that we require neither that the agents' values are drawn from identical distributions, nor that they are independent.

We aim to design VCG redistribution mechanisms that redistribute the most in expectation, subject to the non-deficit constraint. Here, we do not explicitly enforce the individual rationality constraint for the following reasons: Since our objective is to maximize social welfare, if the prior distribution is symmetric across agents, then under any redistribution mechanism that redistributes a nonnegative amount of payment in expectation, every agent benefits from participating in the mechanism (the agent receives nonnegative expected utility). That is, *ex-interim* individual rationality is not a binding constraint. Our technique can also be used to design mechanisms that are ex-interim individually rational when the prior is not symmetric across agents, or mechanisms that satisfy the even stronger *ex-post* individual rationality. However, this would require additional constraints and make the ana-

[8] We remove this restriction in Subsection 2.2.5 where we consider settings without unit demand.

lytical characterization in Subsection 2.2.2 too complex. For the above reasons, we omit individual rationality constraints.

2.2.2 Linear Redistribution Mechanisms

We first restrict our attention to the family of *linear* redistribution mechanisms. We recall from Subsection 2.1.2 that a linear redistribution mechanism is characterized by a linear redistribution function of the following form:

$$r(v_{-i}) = c_0 + c_1 v_{-i,1} + c_2 v_{-i,2} + \ldots + c_{n-1} v_{-i,n-1}$$

Here, $r(v_{-i})$ is agent i's redistribution. $v_{-i,j}$ is the jth highest bid among v_{-i} (the set of bids other than v_i). The coefficients c_j completely characterize the redistribution mechanism.

Optimal-in-expectation linear redistribution mechanisms

We will prove the following result, which characterizes a linear redistribution mechanism that maximizes the expected total redistribution (among linear redistribution mechanisms). We call this mechanism OEL (optimal-in-expectation, linear).

Theorem 6. *Given n, m, and a prior distribution over agents' valuations, the following c_i define a redistribution mechanism that maximizes expected redistribution, under the constraints that the mechanism must be a linear redistribution mechanism, efficient, strategy-proof, and satisfy the non-deficit property.*

Let the o_i be defined as follows:

$$o_0 = U - Ev_1,\ o_i = Ev_i - Ev_{i+1}\ (i = 1, 2, \ldots, n-1),\ \text{and}\ o_n = Ev_n - L.$$

The o_i are determined by the given prior distribution.

Let k be any integer satisfying

$$k \in \arg\min_i \{o_i / \tbinom{n}{i} | i - m \,\text{odd}, i = 0, \ldots, n\}$$

Let function G be defined as follows:

$$G(n,m,i) = \binom{n-i-1}{n-m-1} / \binom{m-1}{i-1} = \frac{(n-i-1)!(i-1)!}{(n-m-1)!(m-1)!}$$

- *If* $0 < k < m$, *then*

  $c_i = (-1)^{m-i} G(n,m,i)$ *for* $i = k+1, \ldots, m$,

  $c_k = m/n - \sum_{i=k+1}^{m} (-1)^{m-i} G(n,m,i)$,

  and $c_i = 0$ *for other* i.

- *If* $k = 0$, *then*

  $c_i = (-1)^{m-i} G(n,m,i)$ *for* $i = 1, \ldots, m$,

  $c_0 = Um/n - U \sum_{i=1}^{m} (-1)^{m-i} G(n,m,i)$,

  and $c_i = 0$ *for other* i.

- *If* $m < k < n$, *then*

  $c_i = (-1)^{m-i-1} G(n,m,i)$ *for* $i = m+1, \ldots, k-1$,

  $c_k = m/n - \sum_{i=m+1}^{k-1} (-1)^{m-i-1} G(n,m,i)$,

  and $c_i = 0$ *for other* i.

- *If* $k = n$, *then*

  $c_i = (-1)^{m-i-1} G(n,m,i)$ *for* $i = m+1, \ldots, n-1$,

  $c_0 = Lm/n - L \sum_{i=m+1}^{n-1} (-1)^{m-i-1} G(n,m,i)$,

  and $c_i = 0$ *for other* i.

In expectation, this mechanism fails to redistribute

$$o_k m \binom{n-1}{m} / \binom{n}{k}$$

This mechanism is uniquely optimal among all linear redistribution mechanisms if and only if the choice of k is unique and there does not exist an even i and an odd j such that $o_i = o_j = 0$.

The mechanism is complicated, and is perhaps easier to understand using the auxiliary variables that we define in the derivation of this mechanism below.

Depending on the optimal choice of k, we have different OEL mechanisms. We call the OEL mechanism corresponding to a specific choice of k the OEL mechanism with index k. For $k = 1, 2, \ldots, n-1$, the waste under the OEL mechanism with index k equals $m\binom{n-1}{m}/\binom{n}{k}(v_k - v_{k+1})$. The waste under the OEL mechanism with index 0 equals $m\binom{n-1}{m}(U - v_1)$. The waste under the OEL mechanism with index n equals $m\binom{n-1}{m}(v_n - L)$. Basically, the waste is always a multiple of: 1) the expected difference between two adjacent (in terms of size) bids, or 2) the expected difference between the upper bound and the largest bid, or 3) the expected difference between the lowest bid and the lower bound. Moreover, the multiplication coefficient is determined by m and n. Then, the OEL mechanism simply chooses the best of these options. In contrast, under the worst-case optimal mechanism, the waste is a linear combination of all of the bids (except for the highest m).

In what follows, we derive the OEL mechanism and prove its optimality. Our objective is to find an linear redistribution mechanism that redistributes the most in expectation. To optimize among the family of linear redistribution mechanisms, we must solve for the optimal values of the c_i. We want the resulting redistribution mechanism to be strategy-proof and efficient, and we want it to satisfy the non-deficit property. The first two properties are satisfied by all the mechanisms inside the linear family, so the only constraint is the non-deficit property. The following optimization model can be used to find the linear redistribution mechanism (the c_i) that redistributes the most in expectation, while satisfying the non-deficit property.

Variables: $c_0, c_1, \ldots, c_{n-1}$
Maximize $E(\sum_{i=1}^{n} r_i)$
Subject to:
For every bid vector $U \geq v_1 \geq v_2 \geq \ldots \geq v_n \geq L$
$\sum_{i=1}^{n} r_i \leq m v_{m+1}$
$r_i = c_0 + c_1 v_{-i,1} + c_2 v_{-i,2} + \ldots + c_{n-1} v_{-i,n-1}$

Given the prior distribution, $E(mv_{m+1})$ is a constant, so the objective of the above model may be rewritten as **Minimize** $E(mv_{m+1} - \sum_{i=1}^{n} r_i)$.

Since $r_i = c_0 + c_1 v_{-i,1} + c_2 v_{-i,2} + \ldots + c_{n-1} v_{-i,n-1}$, where $v_{-i,j}$ is the jth highest bid among bids other than i's own bid, we have the following:

$$r_1 = c_0 + c_1 v_2 + c_2 v_3 + c_3 v_4 \ldots + c_{n-2} v_{n-1} + c_{n-1} v_n$$
$$r_2 = c_0 + c_1 v_1 + c_2 v_3 + c_3 v_4 \ldots + c_{n-2} v_{n-1} + c_{n-1} v_n$$
$$r_3 = c_0 + c_1 v_1 + c_2 v_2 + c_3 v_4 \ldots + c_{n-2} v_{n-1} + c_{n-1} v_n$$
$$\ldots$$
$$r_{n-1} = c_0 + c_1 v_1 + c_2 v_2 + c_3 v_3 \ldots + c_{n-2} v_{n-2} + c_{n-1} v_n$$
$$r_n = c_0 + c_1 v_1 + c_2 v_2 + c_3 v_3 \ldots + c_{n-2} v_{n-2} + c_{n-1} v_{n-1}$$

We can write $mv_{m+1} - \sum_{i=1}^{n} r_i$ as $q_0 + q_1 v_1 + q_2 v_2 + \ldots + q_n v_n$, where the coefficients q_i are listed below:

$$q_0 = -nc_0$$
$$q_i = -(i-1)c_{i-1} - (n-i)c_i \text{ for } i = 1, 2, \ldots, m, m+2, \ldots, n$$
$$q_{m+1} = m - mc_m - (n-m-1)c_{m+1}$$

(We note that we introduced a dummy variable c_n in the above equations—since there are only $n-1$ other bids, c_n will always be multiplied by 0, but adding this variable makes the definition of the q_i more elegant.) Given n and m, $q_0, \ldots, q_n$ ($n+1$ values) are determined by $c_0, \ldots, c_{n-1}$ (n values). Conversely, if $q_0, \ldots, q_{n-1}$ are fixed, then we can completely solve for the values of $c_0, \ldots, c_{n-1}$ (and hence also for q_n). This results in the following relation among the q_i:

$$q_1 - \frac{n-1}{1!} q_2 + \frac{(n-1)(n-2)}{2!} q_3 - \frac{(n-1)(n-2)(n-3)}{3!} q_4 + \ldots + (-1)^{n-1} \frac{(n-1)(n-2)\ldots 2\cdot 1}{(n-1)!} q_n =$$

$$(-1)^m m \frac{(n-1)(n-2)\ldots(n-m)}{m!}$$

After simplification we obtain:

$$\sum_{i=1}^{n} (-1)^{i-1} \binom{n-1}{i-1} q_i = (-1)^m m \binom{n-1}{m}$$

Now, we can use the q_i as the variables of the optimization model, since from them we will be able to infer the c_i. Because $mv_{m+1} - \sum_{i=1}^{n} r_i = q_0 + q_1 v_1 + q_2 v_2 + \ldots + q_n v_n$, we can rewrite the non-deficit constraint by requiring that the latter summation is nonnegative. Also, the q_i must satisfy the previous inequality (otherwise there will be no corresponding c_i).

Variables: $q_0, q_1, \ldots, q_n$
Minimize $E(q_0 + q_1 v_1 + q_2 v_2 + \ldots + q_n v_n)$
Subject to:
For every bid vector $U \geq v_1 \geq v_2 \geq \ldots \geq v_n \geq L$
$q_0 + q_1 v_1 + q_2 v_2 + \ldots + q_n v_n \geq 0$
$\sum_{i=1}^{n} (-1)^{i-1} \binom{n-1}{i-1} q_i = (-1)^m m \binom{n-1}{m}$

In what follows, we will cast the above model into a linear program. We begin with the following lemma:

Lemma 3. *The following are equivalent:*

(1) $q_0 + q_1 v_1 + q_2 v_2 + \ldots + q_n v_n \geq 0$ *for all* $U \geq v_1 \geq v_2 \geq \ldots \geq v_n \geq L$

(2) $q_0 + L \sum_{i=1}^{n} q_i + (U - L) \sum_{i=1}^{k} q_i \geq 0$ *for* $k = 0, \ldots, n$

Proof. (1)⇒(2): (2) can be obtained from (1) by setting $v_1 = v_2 = \ldots = v_k = U$ and $v_{k+1} = v_{k+2} = \ldots = v_n = L$.

(2)⇒(1): Let us rewrite $T = q_0 + q_1 v_1 + q_2 v_2 + \ldots + q_n v_n$ as $q_0 + L \sum_{i=1}^{n} q_i + (v_1 - v_2) \sum_{i=1}^{1} q_i + (v_2 - v_3) \sum_{i=1}^{2} q_i + \ldots + (v_{n-1} - v_n) \sum_{i=1}^{n-1} q_i + (v_n - L) \sum_{i=1}^{n} q_i$. If $\sum_{i=1}^{k} q_i \geq 0$ for every $k = 1, \ldots, n$, then $T \geq q_0 + L \sum_{i=1}^{n} q_i \geq 0$ (because $v_1 - v_2, v_2 -$

$v_3, \ldots, v_n - L$ are all nonnegative). Otherwise, let k' be the index so that $\sum_{i=1}^{k'} q_i$ is minimal (hence negative). To make T minimal, we want $v_{k'} - v_{k'+1}$ (which is multiplied by $\sum_{i=1}^{k'} q_i$) to be maximal. So the minimal value for T is $q_0 + L\sum_{i=1}^{n} q_i + (U-L)\sum_{i=1}^{k'} q_i \geq 0$, which is attained when $v_1 = v_2 = \ldots = v_{k'} = U$ and $v_{k'+1} = v_{k'+2} = \ldots = v_n = L$. Hence T is always nonnegative. □

Let $x_k = (q_0 + L\sum_{i=1}^{n} q_i)/(U-L) + \sum_{i=1}^{k} q_i$ for $k = 0, \ldots, n$. The x_i correspond (one to one) to the q_i, so we can use the x_i as the variables in the optimization model. The first constraint of the optimization model now becomes $x_k \geq 0$ for every k. Since $x_k - x_{k-1} = q_k$ for $k = 1, \ldots, n$, the second constraint becomes

$$\sum_{i=1}^{n}(-1)^{i-1}\binom{n-1}{i-1}(x_i - x_{i-1}) = (-1)^m m\binom{n-1}{m}$$

After simplification we get:

$$\sum_{i=0}^{n}(-1)^i\binom{n}{i}x_i = (-1)^{m-1} m\binom{n-1}{m}$$

Let $o_0 = U - Ev_1$, $o_i = Ev_i - Ev_{i+1}$ $(i = 1, \ldots, n-1)$ and $o_n = Ev_n - L$. The o_i are all nonnegative constants that we know from the prior distribution. The objective of the optimization model can be rewritten as follows:

$$\begin{gathered}
E(q_0 + q_1v_1 + q_2v_2 + \ldots + q_nv_n) \\
= q_0 + q_1Ev_1 + q_2Ev_2 + \ldots + q_nEv_n \\
= x_0(U-L) + q_1(Ev_1 - L) + q_2(Ev_2 - L) + \ldots + q_n(Ev_n - L) \\
= x_0((U-L) - (Ev_1 - L)) + (x_0 + q_1)((Ev_1 - L) - (Ev_2 - L)) + (x_0 + q_1 + \\
q_2)((Ev_2 - L) - (Ev_3 - L)) + \ldots + (x_0 + q_1 + \ldots + q_n)(Ev_n - L) \\
= o_0x_0 + o_1x_1 + \ldots + o_nx_n
\end{gathered}$$

We finally obtain the following linear program:

Variables: $x_0, x_1, \ldots, x_n$
Minimize $o_0 x_0 + o_1 x_1 + \ldots + o_n x_n$
Subject to:
$x_i \geq 0$
$\sum_{i=0}^{n} (-1)^i \binom{n}{i} x_i = (-1)^{m-1} m \binom{n-1}{m}$

At this point, for any given n and m, for any prior distribution, it is possible to solve this linear program using any LP solver; then, using the above, the resulting x_i can be transformed back to c_i to obtain an optimal-in-expectation linear redistribution mechanism. However, this will not be necessary. The following proposition gives an analytical solution of this linear program.

Proposition 20. *Let k be any integer satisfying*

$$k \in \arg\min_i \{o_i / \binom{n}{i} | i - m \,\text{odd}, i = 0, \ldots, n\}$$

The above linear program has the following optimal solution:

$$x_k = m\binom{n-1}{m} / \binom{n}{k}, \text{ and } x_i = 0 \text{ for } i \neq k$$

The optimal objective value is

$$o_k m \binom{n-1}{m} / \binom{n}{k}$$

This solution is the unique optimal solution if and only if the choice of k is unique and there does not exist an even i and an odd j such that $o_i = o_j = 0$.

Proof. We can rewrite the second constraint as

$$\sum_{i=0}^{n} ((-1)^{i-m+1} \binom{n}{i}) / (m \binom{n-1}{m}) x_i = 1$$

This results in the program

Variables: $x_0, x_1, \ldots, x_n$
Minimize $o_0 x_0 + o_1 x_1 + \ldots + o_n x_n$
Subject to:
$x_i \geq 0$
$\sum_{i=0 \ldots n; i-m \,\text{odd}} \binom{n}{i} / (m \binom{n-1}{m}) x_i = \sum_{i=0 \ldots n; i-m \,\text{even}} \binom{n}{i} / (m \binom{n-1}{m}) x_i + 1$

The o_i are nonnegative. To minimize the objective, we want all the x_i to be as small as possible. It is not hard to see that it does not hurt to set the x_i for which $i-m$ is even to zero: in fact, setting them to a larger value will only force the x_i for which $i-m$ is odd to take on larger values, by the last constraint. (It should be noted that if there exists an even i and an odd j such that $o_i = o_j = 0$, then we can increase the corresponding x_i and x_j at no cost to the objective without breaking the constraint, hence the solution is not unique in that case.) This results in the following linear program:

Variables: $x_0, x_1, \ldots, x_n$
Minimize $o_0 x_0 + o_1 x_1 + \ldots + o_n x_n$
Subject to:
$x_i \geq 0$
$\sum_{i=0 \ldots n; i-m \text{ odd}} \binom{n}{i}/(m\binom{n-1}{m})x_i = 1$

We want the x_i to be as small as possible. However, the second constraint makes it impossible to set all the x_i to 0. For each x_i with $i-m$ odd, if we increase it by δ, the left side of the second constraint is increased by $\binom{n}{i}/(m\binom{n-1}{m})\delta$ and the objective value is increased by $o_i\delta$. We need the left side of the second constraint to increase to 1 (starting from 0), while minimizing the increase in the objective value. To do so, we want to find the x_i (with $i-m$ odd) that has the minimal cost-gain ratio (where the cost is $o_i\delta$, and the gain is $\binom{n}{i}/(m\binom{n-1}{m})\delta$). It follows that for any integer k satisfying $k \in \arg\min_i\{o_i/\binom{n}{i} | i-m \text{ odd}, i = 0, \ldots, n\}$, the linear program has the following optimal solution: $x_k = m\binom{n-1}{m}/\binom{n}{k}$ and $x_i = 0$ for $i \neq k$. The resulting optimal objective value is $o_k m\binom{n-1}{m}/\binom{n}{k}$.

In the above argument, there were only two conditions under which we made a choice that is not necessarily uniquely optimal: if (and only if) there exists an even i and an odd j such that $o_i = o_j = 0$, then, as we explained, there exist optimal solutions where some x_i with $m-i$ even is set to a positive value (in fact, it can be set to any value in this case); and if (and only if) $\arg\min_i\{o_i/\binom{n}{i} | i-m \,\text{odd}, i=0,\ldots,n\}$ is not a singleton set, then there exists another optimal solution with another x_k set to a positive value (in fact, in this case, multiple x_k may simultaneously be set to a positive value). □

By transforming the x_i from Proposition 20 to the corresponding c_i, we obtain the OEL mechanism from Theorem 6.

We now present a special case that may give some further intuition. The case where $k = m+1$ in Theorem 6 corresponds to the redistribution mechanism in which each agent receives a redistribution payment that is equal to m/n times the $(m+1)$th highest bid from the other agents. In our setting of multi-unit auctions with unit demand, this is exactly the Bailey-Cavallo mechanism. This observation is formally stated in the following corollary.

Corollary 3. *Given n, m, and a prior distribution over agents' valuations, we define the o_i as follows:*

$$o_0 = U - Ev_1,\ o_i = Ev_i - Ev_{i+1}\ (i = 1, 2, \ldots, n-1),\ \text{and}\ o_n = Ev_n - L.$$

If the following condition holds:

$$o_{m+1} \le o_i\binom{n}{m+1}/\binom{n}{i}\ \text{for all}\ 0 \le i \le n\ \text{with}\ i - m\ \text{odd},$$

then the Bailey-Cavallo mechanism maximizes expected redistribution, under the constraints that the mechanism must be a linear redistribution mechanism, efficient, strategy-proof, and satisfy the non-deficit property.

Next, we present two example OEL mechanisms.

Example 3. Consider the case where $n = 3$ and $m = 1$, and the bids are all drawn independently and uniformly from $[0, 1]$. In this case, $Ev_i = \frac{4-i}{4}$ for $i = 1, \ldots, 3$. So, $U = 1, L = 0, o_i = \frac{1}{4}$ for $i = 0, \ldots, 3$. (We recall that $o_0 = U - Ev_1, o_n =$

$Ev_n - L$, and $o_i = Ev_i - Ev_{i+1}$ otherwise.) $\arg\min_i\{o_i/\binom{n}{i} | i - m \text{ odd}, i = 0, \ldots, n\}$ is then $\{m+1\} = \{2\}$. The expected amount that fails to be redistributed is $o_2 m\binom{n-1}{m}/\binom{n}{2} = \frac{1}{6}$. (The expected total VCG payment is $\frac{1}{2}$.) The optimal solution is given by $c_2 = \frac{1}{3}$, and $c_i = 0$ for other i. Hence, this optimal-in-expectation linear redistribution mechanism is defined by $r_i = \frac{1}{3}v_{-i,2}$, which is actually the Bailey-Cavallo mechanism. The total redistribution is $\sum_{i=1}^n r_i = \frac{1}{3}v_2 + \frac{2}{3}v_3$. The expected amount that fails to be redistributed is $E(v_2 - \frac{1}{3}v_2 - \frac{2}{3}v_3) = \frac{2}{3}E(v_2 - v_3) = \frac{1}{6}$.

Example 4. Consider the case where $n = 8$ and $m = 2$, and the bids are all drawn independently and uniformly from $[0, 1]$. In this case, $Ev_i = \frac{9-i}{9}$ for $i = 1, \ldots, 8$. So $U = 1, L = 0, o_i = \frac{1}{9}$ for $i = 0, \ldots, 8$. $\arg\min_i\{o_i/\binom{n}{i} | i - m \text{ odd}, i = 0, \ldots, n\}$ is then $\{3, 5\}$. The expected amount that fails to be redistributed is $o_3 m\binom{n-1}{m}/\binom{n}{3} = \frac{1}{12}$. (The expected total VCG payment is $\frac{4}{3}$.)

One optimal solution is given by $c_3 = \frac{1}{4}$, and $c_i = 0$ for other i. Hence this expectation optimal linear redistribution mechanism is defined by $r_i = \frac{1}{4}v_{-i,3}$ (Bailey-Cavallo mechanism). The total redistribution is $\sum_{i=1}^n r_i = \frac{5}{4}v_3 + \frac{3}{4}v_4$. The expected amount that fails to be redistributed is $E(2v_3 - \frac{5}{4}v_3 - \frac{3}{4}v_4) = \frac{3}{4}E(v_3 - v_4) = \frac{1}{12}$.

The other optimal solution is given by $c_3 = \frac{2}{5}, c_4 = -\frac{3}{10}, c_5 = \frac{3}{20}$, and $c_i = 0$ for other i. Hence this expectation optimal linear redistribution mechanism is defined by $r_i = \frac{2}{5}v_{-i,3} - \frac{3}{10}v_{-i,4} + \frac{3}{20}v_{-i,5}$. The total redistribution is $\sum_{i=1}^n r_i = 2v_3 - \frac{3}{4}v_5 + \frac{3}{4}v_6$. The expected amount that fails to be redistributed is $E(\frac{3}{4}(v_5 - v_6)) = \frac{1}{12}$.

Properties of the OEL mechanism

We present some properties of the OEL mechanism. First, we have that there cannot be another redistribution mechanism that always redistributes at least as much *in total* as OEL, and strictly more in at least one case. That is, the OEL mechanism is *collectively undominated*.

Proposition 21. *For any m, n and any prior distribution, there does not exist any redistribution mechanism that, for every multi-set of bids, redistributes at least as much in total as OEL, and redistributes strictly more in at least one case.*

We will prove the above proposition in Section 2.3. More precisely, we will show that the OEL mechanisms characterized in Theorem 6 are the only collectively undominated redistribution mechanisms that are anonymous and linear in multi-unit auctions with unit demand.

It should be noted that Proposition 21 only applies to the OEL mechanism, as defined in Theorem 6. Under certain circumstances (as detailed in Theorem 6), this mechanism is not uniquely optimal; and the other optimal mechanisms do not always have the property of Proposition 21.

The next proposition shows that, if the prior distribution does not distinguish among agents, OEL is *ex-interim* individually rational—that is, in expectation, agents benefit from participating in the mechanism (they receive nonnegative expected utilities).

Proposition 22. *If the prior distribution is symmetric across agents (for example, the agents' values are independent and identically distributed), then the OEL redistribution mechanism is ex-interim individually rational.*

Proof. The original VCG mechanism (redistributing nothing) is also a linear redistribution mechanism (corresponding to $c_i = 0$ for all i). Hence, the OEL mechanism will always redistribute a nonnegative amount in expectation. That is, $E(\sum_{i=1}^{n} r_i) \geq 0$. If the distribution is symmetric across agents, $E(r_i) = E(r_j)$ for any i and j ($E(r_i)$ is the expected redistribution received by agent i, which is independent of her own report). So $E(r_i) \geq 0$ for all i. However, the VCG mechanism is well-known to be ex-interim (in fact, ex-post) individually rational in this setting, so that even if

$E(r_i) = 0$, agents' expected utility from participating in the mechanism is nonnegative. It follows that OEL must also be ex-interim individually rational. □

As an aside, if the prior is not symmetric across agents, then we can explicitly add the ex-interim individual rationality constraint (or the stronger *ex-post* individual rationality constraint[9]) into our optimization model. This still results in a linear program (but it does not admit an elegant analytical solution).

In Theorem 6, we gave an expression for the expected amount that OEL fails to redistribute, which depended on the prior. In the next proposition, we give an upper bound on this that does not depend on the prior.

Proposition 23. *For any prior, the OEL mechanism fails to redistribute at most*

$$(U-L)m\binom{n-1}{m} / \sum_{i=0,1,\ldots,n; i-m \text{ odd}} \binom{n}{i}$$

in expectation. This bound is tight.

Proof. Given a prior distribution (and therefore, given the o_i), the expected amount that fails to be redistributed is $o_k m\binom{n-1}{m}/\binom{n}{k}$ for any $k \in \arg\min_i\{o_i/\binom{n}{i} | i-m \text{ odd}, i = 0,\ldots,n\}$. If a distribution is constructed such that $o_i = (U-L)\binom{n}{i}/\sum_{i=0,\ldots,n; i-m \text{ odd}} \binom{n}{i}$ for all i with $i-m$ odd, and $o_i = 0$ for all other i (this is in fact a feasible setting of the o_i—we can just use a degenerate distribution where the agents' valuations are not independent), then $\arg\min_i\{o_i/\binom{n}{i} | i-m \text{ odd}, i = 0,\ldots,n\} = \{i | 0 \le i \le n, i-m \text{ odd}\}$. So k can be any i as long as $i-m$ is odd. In this case, the expected amount not redistributed is exactly $(U-L)m\binom{n-1}{m}/\sum_{i=0,\ldots,n; i-m \text{ odd}} \binom{n}{i}$.

Now suppose that there is another distribution under which the mechanism fails to redistribute strictly more in expectation. Then, the new set of o'_i must satisfy

[9] A mechanism is ex-post individually rational if every agent receives nonnegative utility for *any* bids.

$o'_i m\binom{n-1}{m}/\binom{n}{k} > m\binom{n-1}{m}/\sum_{i=0,\dots,n; i-m \text{ odd}} \binom{n}{i} = o_i m\binom{n-1}{m}/\binom{n}{k}$ for any i with $i-m$ odd. It follows that $o'_i > o_i$ for any i with $i-m$ odd. Since $\sum_{i=0,\dots,n; i-m \text{ odd}} o_i = U - L$ and $o'_i \geq 0$ for any i with $i-m$ even, we have $\sum_{i=0,\dots,n} o'_i > U - L$, which is a contradiction. □

For Example 3, Proposition 23 gives an upper bound on the expected amount that fails to be redistributed of 0.5 (we recall that the actual amount is $\frac{1}{6}$). For Example 4, Proposition 23 gives an upper bound on the expected amount that fails to be redistributed of 0.3281 (we recall that the actual amount is $\frac{1}{12}$).

The next proposition shows that for fixed m, as n goes to infinity, the expected amount that fails to be redistributed goes to 0; hence OEL is asymptotically optimal for fixed number of units.

Proposition 24. *For fixed m, as n goes to infinity, the expected amount that fails to be redistributed by OEL goes to* 0.

Proof. By Proposition 23, we only need to show that for fixed m, as n goes to infinity, $(U-L)m\binom{n-1}{m}/\sum_{i=0,1,\dots,n; i-m \text{ odd}} \binom{n}{i}$ goes to 0.

We have that $(U-L)m\binom{n-1}{m}/\sum_{i=0,1,\dots,n; i-m \text{ odd}} \binom{n}{i} \leq (U-L)m\binom{n-1}{m}/\binom{n}{m+1} = (U-L)\frac{m(n-1)!(m+1)!(n-m-1)!}{m!(n-m-1)!n!} = (U-L)(m+1)m/n$. The right-hand side goes to 0 as n goes to infinity. □

On the other hand, if we increase both n and m, and keep their difference within constant C, then the expected amount fails to redistributed by OEL also goes to 0: for large n, the expected amount fails to be redistributed by OEL is at most $(U-L)m\binom{n-1}{m}/\sum_{i=0,1,\dots,n; i-m \text{ odd}} \binom{n}{i} \leq (U-L)n(n-1)^{C-1}/\sum_{i=0,1,\dots,n; i-m \text{ odd}} \binom{n}{i}$

$= (U-L)n(n-1)^{C-1}/(\sum_{i=1,2,\ldots,n-1; i-m \text{ odd}}(\binom{n-1}{i-1}+\binom{n-1}{i})+\sum_{i=0,n; i-m \text{ odd}}\binom{n}{i})$. Basically, the denominator is exponential in n, while the numerator is polynomial in n. Therefore, as n increases, the amount fails to be redistributed by OEL approaches 0.

So far, we have only considered anonymous redistribution mechanisms (that is, mechanisms with the same redistribution function $r(\cdot)$ for each agent).[10] If we allow the redistribution mechanism to be nonanonymous, then we can use different c_i for different bidders. Moreover, even for the same bidder, we can use different c_i depending on the order of the other bidders (in terms of their bids), and there are $(n-1)!$ such orders. Thus, it is clear that to optimize among the class of nonanonymous linear redistribution mechanisms, we need significantly more variables, and analytical solution of the linear program no longer seems tractable. However, we do have the following proposition, which shows that the OEL mechanism remains optimal even among nonanonymous linear redistribution mechanisms, if the prior is symmetric.

Proposition 25. *If the prior distribution is symmetric across agents (for example, the agents' values are independent and identically distributed), then no nonanonymous linear redistribution mechanism can redistribute strictly more than the OEL mechanism (which is anonymous) in expectation.*

Proof. Let us define the average of two (not necessarily anonymous) redistribution mechanisms as follows: for any multi-set of bids, for any agent i, if one redistribution mechanism redistributes x to agent i, and the other redistribution mechanism redistributes y to i, then the average mechanism redistributes $(x+y)/2$ to i. It is not difficult to see that if two redistribution mechanisms both never incur a deficit, then the average of these two mechanisms also satisfies the non-deficit property. This averaging operation is easily generalized to averaging over three or more mechanisms.

[10] An exception is Proposition 21, which shows that there is not even a nonanonymous mechanism that always redistributes at least as much in total as OEL, and strictly more in at least one case.

Now let us assume that r is a nonanonymous linear redistribution mechanism, and that r redistributes strictly more than the OEL mechanism in expectation when the prior distribution is symmetric across agents. Let π be any permutation of n elements. We permute the way r treats the agents according to π, and denote the new mechanism by r^π. That is, r^π treats agent $\pi(i)$ the way r treats i. Since we assumed that the prior distribution is symmetric across agents, the expected total amount redistributed by r^π should be the same as that redistributed by r. Now, if we take the average of the r^π over all permutations π, we obtain an anonymous linear redistribution mechanism that redistributes as much in expectation as r (and hence more than the OEL mechanism). But this contradicts the optimality of the OEL mechanism among anonymous linear redistribution mechanisms. □

2.2.3 Discretized Redistribution Mechanisms

In the previous subsection, we only considered linear redistribution mechanisms. This restriction allowed us to find the optimal linear redistribution mechanism by analytically solving a linear program. In this subsection, we consider a larger domain of eligible mechanisms, and propose *discretized redistribution mechanisms*, which can be automatically designed [30] and can outperform the OEL mechanism. (In this subsection, for simplicity and to be able to compare to the previous subsection, we only consider anonymous mechanisms, and we do not impose an individual rationality constraint. However, all of the below can be generalized to allow for nonanonymous mechanisms and an individual rationality constraint.)

We study the following problem: given a prior distribution f (the joint pdf of $v_1, v_2, \ldots, v_n$), we try to find a redistribution mechanism that redistributes the most in expectation among all redistribution mechanisms that can be characterized by continuous functions. For simplicity, we will assume that f is continuous. The optimization model is the following:

Variable function: $r : \mathbf{R}^{n-1} \to \mathbf{R}$, r continuous
Maximize
$\int_{U \geq v_1 \geq \ldots \geq v_n \geq L} \sum_{i=1}^{n} r(v_{-i}) f(v_1, v_2, \ldots, v_n) dv_1 dv_2 \ldots dv_n$
Subject to:
For every bid vector $U \geq v_1 \geq v_2 \geq \ldots \geq v_n \geq L$
$\sum_{i=1}^{n} r(v_{-i}) \leq m v_{m+1}$

Let R^* be the optimal objective value for this model. (To be precise, we have not proved that an optimal solution exists for this model: it could be that the set of feasible solution values does not include its least upper bound. In this case, simply let R^* be the least upper bound.) Since we are not able to solve this model analytically, we try to solve it numerically.

We divide the interval $[L, U]$ (within which the bids lie) into N equal parts, with step size $h = (U - L)/N$. Let k denote the subinterval: $I(k) = [L + kh, L + kh + h]$ $(k = 0, 1, \ldots, N - 1)$. Define $r^h : \mathbf{R}^{n-1} \to \mathbf{R}$ as follows: for all $U \geq x_1 \geq x_2 \geq \ldots \geq x_{n-1} \geq L$, $r^h(x_1, x_2, \ldots, x_{n-1}) = z^h[k_1, k_2, \ldots, k_{n-1}]$ where $k_i = \lfloor (x_i - L)/h \rfloor$ (except that $k_i = N - 1$ if $x_i = U$). Here, the $z^h[k_1, k_2, \ldots, k_{n-1}]$ are variables. We call such a mechanism a discretized redistribution mechanism of step size h.

Proposition 26. *A discretized redistribution mechanism satisfies the non-deficit constraint if and only if*

$$\sum_{i=1}^{n} z^h[k_1, k_2, \ldots, k_{i-1}, k_{i+1}, \ldots, k_n] \leq m(L + k_{m+1} h)$$
$$\text{for every } N - 1 \geq k_1 \geq k_2 \geq \ldots \geq k_n \geq 0.$$

Proof. For every $N - 1 \geq k_1 \geq k_2 \geq \ldots \geq k_n \geq 0$, if $v_i = L + k_i h$ for all i, then $\sum_{i=1}^{n} z^h[k_1, k_2, \ldots, k_{i-1}, k_{i+1}, \ldots, k_n]$ is the total redistribution and $m(L + k_{m+1} h)$ is the total VCG payment. It follows that if the mechanism satisfies the non-deficit property, the inequalities in the proposition must hold. Conversely, if all the inequalities in the proposition hold, then the total redistribution of the mechanism is never more than $m(L + k_{m+1}h)$, which is less than equal to the total VCG payment mv_{m+1}. So the mechanism never incurs a deficit if all the inequalities in the proposition hold. □

The following linear program finds the optimal discretized redistribution mechanism for step size h. The variables are $z^h[k_1, k_2, \ldots, k_{n-1}]$ for all integers k_i satisfying $N-1 \geq k_1 \geq k_2 \geq \ldots \geq k_{n-1} \geq 0$. The objective is the expected total redistribution, where $p[k_1, k_2, \ldots, k_n] = P(v_1 \in I(k_1), v_2 \in I(k_2), \ldots, v_n \in I(k_n))$ (we note that the $p[k_1, k_2, \ldots, k_n]$ are constants).

Variables: $z^h[\ldots]$
Maximize
$\sum_{N-1\geq k_1\geq k_2\geq\ldots\geq k_n\geq 0} p[k_1, k_2, \ldots, k_n] \sum_{i=1}^{n} z^h[k_1, k_2, \ldots, k_{i-1}, k_{i+1}, \ldots, k_n]$
Subject to:
For every $N-1 \geq k_1 \geq k_2 \geq \ldots \geq k_n \geq 0$
$\sum_{i=1}^{n} z^h[k_1, k_2, \ldots, k_{i-1}, k_{i+1}, \ldots, k_n] \leq m(L + k_{m+1}h)$

Let $z^{*h}[\ldots]$ denote the optimal solution of the above linear program, and let r^{*h} denote the corresponding optimal discretized redistribution mechanism. Let R^{*h} denote the optimal objective value. The next proposition shows that discretized redistribution mechanisms cannot outperform the best continuous redistribution mechanisms.

Proposition 27. $R^{*h} \leq R^*$.

Proof. For any $\epsilon > 0$, we will show how to construct a continuous function r'_ϵ so that $r'_\epsilon \leq r^{*h}$ everywhere, and the measure of the set $\{r^{*h} \neq r'_\epsilon\}$ is less than ϵ.

Let B be the greatest lower bound of r^{*h} (r^{*h} is bounded below because it is a piecewise constant function with finitely many pieces). For given $U \geq x_1 \geq x_2 \geq \ldots \geq x_{n-1} \geq L$, let $d(x_1, \ldots, x_{n-1})$ be the minimal distance from any $x_i - L$ to the nearest multiple of h. For any $\delta > 0$, let $r_\delta(x_1, \ldots, x_{n-1}) = r^{*h}(x_1, \ldots, x_{n-1})$ if

$d(x_1, \ldots, x_{n-1}) > \delta$, and $r_\delta(x_1, \ldots, x_{n-1}) = r^{*h}(x_1, \ldots, x_{n-1})$
$- (\delta - d(x_1, \ldots, x_{n-1}))(r^{*h}(x_1, \ldots, x_{n-1}) - B)/\delta$ otherwise.

It is easy to see that the function r_δ is continuous at any point where $d(x_1, \ldots, x_{n-1}) > \delta$, because at these points, r^{*h} is continuous. Furthermore, the function is continuous at any point where $\delta > d(x_1, \ldots, x_{n-1}) > 0$, because r^{*h} and d are both continuous at these points. Moreover, it is also continuous at any point where $d(x_1, \ldots, x_{n-1}) = \delta$, because at such a point $r^{*h}(x_1, \ldots, x_{n-1}) - (\delta - d(x_1, \ldots, x_{n-1}))(r^{*h}(x_1, \ldots, x_{n-1}) - B)/\delta = r^{*h}(x_1, \ldots, x_{n-1})$. Finally, at any point where $d(x_1, \ldots, x_{n-1}) = 0$, the function is continuous because on any point $x'_1, \ldots, x'_{n-1}$ at distance at most $\gamma > 0$ from $x_1, \ldots, x_{n-1}$, the function will take value at most $\gamma(H - B)/\delta$, where H is an upper bound on r^{*h} (H is finite).

As δ goes to 0, so does the measure of the set $\{r^{*h} \neq r_\delta\}$. Moreover, $r_\delta \leq r^{*h}$ everywhere. Hence we can obtain r'_ϵ with the desired property by letting it equal r_δ for sufficiently small δ.

Now, r'_ϵ is a feasible redistribution mechanism, because it always redistributes less than r^{*h}. Moreover, because f is a continuous pdf on a compact domain, as $\epsilon \to 0$, the difference in expected value between r'_ϵ and r^{*h} goes to 0. Hence, we can create continuous redistribution functions that come arbitrarily close to R^{*h} in terms of expected redistribution, and hence R^* (the least upper bound of the expected redistributions that can be obtained with continuous functions) is at least R^{*h}. □

The next proposition shows that if we make the discretization finer, we will do no worse.

Proposition 28. $R^{*h} \leq R^{*h/2}$.

Proof. For all $2N - 1 \geq k_1 \geq k_2 \geq \ldots \geq k_{n-1} \geq 0$, let $z^{h/2}[k_1, k_2, \ldots, k_{n-1}] = z^{*h}[\lfloor k_1/2 \rfloor, \lfloor k_2/2 \rfloor, \ldots, \lfloor k_{n-1}/2 \rfloor]$. The discretized redistribution mechanism corresponding to $z^{h/2}[\ldots]$ is exactly r^{*h}. The discretized redistribution mechanism r^{*h} satisfies the non-deficit property. Hence the variables $z^{h/2}[\ldots]$ form a feasible solution of the linear program corresponding to step size $h/2$, so its expected redistribution must be less than or equal to that of the optimal solution of the linear program corresponding to step size $h/2$. That is, $R^{*h} \leq R^{*h/2}$. □

The next proposition shows that as we make the discretization finer and finer, we converge to the optimal value for continuous redistribution mechanisms.

Proposition 29. $\lim_{h\to 0} R^{*h} = R^*$.

Proof. For any $\gamma > 0$, there exists a continuous redistribution mechanism r^* such that its expected redistribution is at least $R^* - \gamma$. r^* is continuous on a closed and bounded domain, so r^* is uniformly continuous. Hence for any $\epsilon > 0$, there exists $\delta > 0$ so that $|r^*(x_1, x_2, \ldots, x_{n-1}) - r^*(x'_1, x'_2, \ldots, x'_{n-1})| \leq \epsilon$ as long as $\max_i\{|x_i - x'_i|\} \leq \delta$. Choose $h \leq \delta$, and define $z^h[k_1, k_2, \ldots, k_{n-1}]$ by $r^*(L + k_1 h, L + k_2 h, \ldots, L + k_{n-1} h)$ for all $N - 1 \geq k_1 \geq k_2 \geq \ldots \geq k_{n-1} \geq 0$. $z^h[\ldots]$ corresponds to a feasible discretized mechanism r^h. In addition, $r^h \geq r^* - \epsilon$. Hence, the expected redistribution of the optimal discretized mechanism with step size (at most) h is $R^{*h} \geq R^* - \gamma - n\epsilon$. Since γ and ϵ are both arbitrarily small, $\lim_{h\to 0} R^{*h} \geq R^*$. By Proposition 27, $\lim_{h\to 0} R^{*h} \leq R^*$. □

We note that a discretized redistribution mechanism r^h is defined by a finite number of real-valued variables: namely, one variable $z^h[k_1, k_2, \ldots, k_{n-1}]$ for every $N - 1 \geq k_1 \geq k_2 \geq \ldots \geq k_{n-1} \geq 0$. Because of this, we can use a standard LP solver to solve for the optimal discretized redistribution mechanism r^h (for given m, n, h and prior). In general, this linear program involves exponential number of variables and does not scale. However, at least for small problem instances, we can set h to very small values, and by Proposition 29, we expect the resulting mechanism to be close to optimal.

But how do we know how far from optimal we are? As it turns out, the discretization method can also be used to find upper bounds on R^*. Here, we will assume that agents' values are independent and identically distributed. The following linear program gives an upper bound on R^*.

Variables: $z^h[\ldots]$
Maximize
$\sum_{N-1 \geq k_1 \geq k_2 \geq \ldots \geq k_n \geq 0} p[k_1, k_2, \ldots, k_n] \sum_{i=1}^{n} z^h[k_1, k_2, \ldots, k_{i-1}, k_{i+1}, \ldots, k_n]$
Subject to:
For every $N-1 \geq k_1 \geq k_2 \geq \ldots \geq k_n \geq 0$
$\sum_{i=1}^{n} z^h[k_1, k_2, \ldots, k_{i-1}, k_{i+1}, \ldots, k_n] \leq$
$mE(v_{m+1} | v_1 \in I(k_1), v_2 \in I(k_2), \ldots, v_n \in I(k_n))$

The intuition behind this linear program is the following. In the previous linear program, the non-deficit constraints were effectively set for the *lowest* values within each discretized block, which guaranteed that they would hold for every value in the block. In this linear program, however, we set the non-deficit constraints by taking the *expectation* over the values in each block. Generally, this will result in deficits for values inside the block, so this program does not produce feasible mechanisms.

Let $\hat{z}^h[\ldots]$ denote the optimal solution of the above linear program, and let $\hat{r}^h$ denote the (not necessarily feasible) corresponding optimal discretized redistribution mechanism. Let $\hat{R}^h$ denote the optimal objective value. We have the following propositions:

Proposition 30. *If the bids are independent and identically distributed, then $\hat{R}^h \geq R^*$.*

Proof. Let r be any feasible continuous (anonymous) redistribution mechanism. Now, consider the conditional expectation of a bidder's redistribution payment under r, given that, for each $i \in \{1, \ldots, n-1\}$, the ith highest bid among other bidders is in $I(k_i) = [L + k_i h, L + k_i h + h]$. Let $z^h[k_1, k_2, \ldots, k_{n-1}]$ denote this conditional expectation. (We emphasize that this does not depend on which agent we choose, due to the i.i.d. assumption.)

Now, these $z^h[\ldots]$ constitute a feasible solution of the above linear program, for the following reason. The left-hand side of the constraint in the above linear program is now the expected total redistribution of r, given that for each $i \in \{1, \ldots, n\}$, the ith highest bid is in $I(k_i)$; and the right-hand side is the expected total VCG payment, given that for each $i \in \{1, \ldots, n\}$, the ith highest bid is in $I(k_i)$. Because r satisfies non-deficit by assumption, the constraint must be met by the $z^h[\ldots]$.

Moreover, the objective value of the feasible solution defined by the $z^h[\ldots]$ is identical to the expected total amount redistributed by r. Hence, for every expected total amount redistributed by a feasible continuous mechanism, there exists a feasible solution to the above linear program that attains that value. It follows that $\hat{R}^h \geq R^*$. □

So, we have that R^{*h} is a lower bound on R^*, and $\hat{R}^h$ is an upper bound. The next proposition considers how close these two bounds are, in terms of the step size h.

Proposition 31. *If the bids are independent and identically distributed, then* $\hat{R}^h \leq R^{*h} + mh$.

Proof. Consider the right-hand side of the constraints of the above linear program. We have $mE(v_{m+1}|v_1 \in I(k_1), v_2 \in I(k_2), \ldots, v_n \in I(k_n)) \leq m(L + k_{m+1}h + h)$, since $v_{m+1} \in I(k_{m+1})$ implies that $v_{m+1} \leq L + k_{m+1}h + h$. Consider an optimal solution of the linear program for determining $\hat{R}^h$. Now, from every variable $z^h[k_1, k_2, \ldots, k_{n-1}]$, subtract mh/n. This results in a feasible solution of the linear program for determining R^{*h}, and the decrease in the objective value is $nmh/n = mh$. Hence, $\hat{R}^h \leq R^{*h} + mh$. □

Hence, by solving the linear program for determining R^{*h}, we get a lower bound on R^* and a discretized redistribution mechanism that comes close to it. If we also have that the bids are independent and identically distributed, by solving the linear program for determining $\hat{R}^h$, we get an upper bound on R^* that is close to R^{*h}.

2.2.4 Experimental Results

We now have two different types of redistribution mechanisms with which we can try to maximize the expected total amount redistributed. The OEL mechanism has the advantage that Theorem 6 gives a simple expression for it, so it is easy to scale to large auctions. In addition, it is optimal among all linear redistribution mechanisms, although nonlinear redistribution mechanisms may perform even better in expectation despite not being able to welfare dominate the OEL mechanism. On the other hand, the discretized mechanisms have the advantage that, as we decrease the step size h, we will converge to the maximum amount that can be redistributed by any continuous redistribution mechanism. The disadvantage of this approach is that it does not scale to large auctions. Fortunately, the experimental results below show that for auctions with many bidders, the OEL mechanism redistributes almost everything, whereas for auctions with few bidders, we can solve for the optimal discretized redistribution mechanism with a very small step size. That is, the two types of redistribution mechanisms are in some sense complementary.

In Table 2.3, for different n (number of agents) and m (number of units), we list the expected amount of redistribution by both the OEL mechanism and the optimal discretized mechanism (for specific step sizes). The bids are independently drawn from the uniform $[0, 1]$ distribution.

In Table 2.3, the column "VCG" gives the expected total VCG payment; the column "BC" gives the expected redistribution by the Bailey-Cavallo mechanism; the column "OEL" gives the expected redistribution by the OEL mechanism; the

Table 2.3: Expected redistribution by VCG, BC, OEL, and discretized mechanisms, for small numbers of agents.

n,m	VCG	BC	OEL	R^{*h}	$\hat{R}^h$	%
3,1	0.5000	0.3333	0.3333	0.4218 (N=100)	0.4269	84.4
4,1	0.6000	0.5000	0.5000	0.5498 (N=40)	0.5625	91.6
5,1	0.6667	0.6000	0.6000	0.6248 (N=25)	0.6452	93.7
6,1	0.7143	0.6667	0.6667	0.6701 (N=15)	0.7040	93.8
3,2	0.5000	0.0000	0.3333	0.4169 (N=100)	0.4269	83.4
4,2	0.8000	0.5000	0.5000	0.6848 (N=40)	0.7103	85.6
5,2	1.0000	0.8000	0.8000	0.8944 (N=25)	0.9355	89.4
6,2	1.1429	1.0000	1.0000	1.0280 (N=15)	1.0978	89.9

Table 2.4: Expected redistribution by VCG, BC, and OEL for large numbers of agents.

n,m	VCG	BC	OEL	%	n,m	VCG	BC	OEL	%
10,1	0.8182	0.8000	0.8143	99.5	20,1	0.9048	0.9000	0.9048	100.0
10,3	1.9091	1.8000	1.8000	94.3	20,5	3.5714	3.5000	3.5564	99.6
10,5	2.2727	2.0000	2.0000	88.0	20,10	4.7619	4.5000	4.5000	94.5
10,7	1.9091	1.4000	1.8000	94.3	20,15	3.5714	3.0000	3.5564	99.6
10,9	0.8182	0.0000	0.8143	99.5	20,19	0.9048	0.0000	0.9048	100.0

column "R^{*h}" gives the expected redistribution by the optimal discretized redistribution mechanism (step size $1/N$); the column "$\hat{R}^h$" gives the upper bound on the expected redistribution by any continuous redistribution mechanism (same step size as that of R^{*h}). The last column gives the percentages of the VCG payment that are redistributed by the optimal discretized redistribution mechanisms (rounding to the nearest tenth).

Finally, when the number of agents is large, the OEL mechanism is very close to optimal, as shown in Table 2.4:

The fifth and tenth columns give the percentages of the VCG payment that are redistributed by the OEL mechanisms (rounding to the nearest tenth).

2.2.5 Multi-Unit Auctions with Nonincreasing Marginal Values

In this subsection, we consider a more general setting in which agents do not necessarily have unit demand, that is, they may value receiving units in addition to the first. However, we assume that the marginal values are nonincreasing, that is, they value the earlier units (weakly) more. (Units remain indistinguishable.) We still use n and m to denote the number of agents and the number of available units, but we no longer require that $m < n$. An agent's bid is now a nonincreasing sequence of m elements. We denote agent i's bid by $B_i =< b_{i1}, b_{i2}, \ldots, b_{im} >$, where b_{ij} is agent i's marginal value for getting her jth unit (so that $b_{ij} \geq b_{i(j+1)}$). That is, agent i's valuation for receiving j units is $\sum_{k=1}^{j} b_{ik}$. A bid profile now consists of n vectors B_i, with $1 \leq i \leq n$, or equivalently mn elements b_{ij}, with $1 \leq i \leq n$ and $1 \leq j \leq m$. We represent the b_{ij} in matrix form as follows:

$$\begin{bmatrix} b_{1m} & b_{2m} & \ldots & b_{nm} \\ \ldots & \ldots & \ldots & \ldots \\ b_{12} & b_{22} & \ldots & b_{n2} \\ b_{11} & b_{21} & \ldots & b_{n1} \end{bmatrix}$$

Without loss of generality, we assume that $b_{11} \geq b_{21} \geq \ldots \geq b_{n1}$. That is, the agents are ordered according to their marginal values for winning their first unit. We denote the kth highest element among all the b_{ij} by v_k ($1 \leq k \leq mn$).

We assume that we know the joint distribution of the b_{ij} (and hence we also know the joint distribution of the v_k). We continue to use U to denote the known upper bound on the values that the b_{ij} can take (U is also the upper bound on the v_k). In this subsection we will not consider the case where there is a lower bound $L > 0$ on all the b_{ij} (v_k); that is, we assume the lower bound is 0. (In fact, if there is a lower bound $L > 0$, we can simply require the agents to bid how far above L their marginal values are, that is, require them to submit $b'_{ij} = b_{ij} - L$, in which case we arrive at the case that we study below. The VCG payments under these modified bids will always be mL less than under the original bids, but we can easily redistribute this additional mL. Hence, the restriction that $L = 0$ comes without loss of generality.)

Let B be a bid profile. We denote the set of bids other than B_i (agent i's own bid) by B_{-i}. B_{-i} consists of $mn - m$ elements. We can write B_{-i} in matrix form as follows:

$$\begin{bmatrix} b_{1m} & \dots & b_{i-1,m} & b_{i+1,m} & \dots & b_{nm} \\ \dots & \dots & \dots & \dots\dots & \dots & \dots \\ b_{12} & \dots & b_{i-1,2} & b_{i+1,2} & \dots & b_{n2} \\ b_{11} & \dots & b_{i-1,1} & b_{i+1,1} & \dots & b_{n1} \end{bmatrix}$$

We denote the kth highest element in B_{-i} by $v_{-i,k}$ $(1 \le k \le mn - m)$.

Our definition for VCG redistribution mechanisms in this setting is similar to our earlier definition. Namely, in a VCG redistribution mechanism, we first allocate the units efficiently, according to the VCG mechanism; then, each agent receives a redistribution payment that is independent of her own bid. An efficient allocation is obtained by accepting the m highest marginal values $(v_1, v_2, \dots, v_m)$. That is, if x elements among $v_1, v_2, \dots, v_m$ come from agent i's bid, then agent i wins x units. Agent i's redistribution equals $r(B_{-i})$, where r is the function that characterizes the redistribution rule.

We now need a definition of *linear* redistribution mechanisms in this setting. We could define linear redistribution mechanisms as follows:

$$r(B_{-i}) = c_0 + c_1 v_{-i,1} + c_2 v_{-i,2} + \dots + c_{mn-m} v_{-i,mn-m}$$

We will study this particular definition later; however, it should immediately be noted that this definition ignores some potentially valuable information in B_{-i}, as shown by the following example.

Example 5. Let $n = 3$ and $m = 2$.

- Case 1: Let B_{-i} be $\begin{bmatrix} 0 & 0 \\ U & U \end{bmatrix}$.

- Case 2: Let B_{-i} be $\begin{bmatrix} U & 0 \\ U & 0 \end{bmatrix}$.

In both cases, we have $v_{-i,1} = U$, $v_{-i,2} = U$, $v_{-i,3} = 0$, and $v_{-i,4} = 0$. Hence, if we define the linear redistribution mechanisms as above, then the redistribution payment must be the same in both cases.

We can see that the above definition loses some information about the ordering of the elements in the matrix. We will show later that this information loss can in fact come at a cost (less payments can be redistributed). It would be good if we can incorporate the information about the order of the b_{ij} in B_{-i} in the definition of linear redistribution mechanisms. This is what we will do next.

Let B and B' be two bid profiles. The elements in B and B' are denoted by b_{ij} and b'_{ij}, respectively, for $1 \leq i \leq n$ and $1 \leq j \leq m$. We say B and B' are **order consistent**, denoted by $B \simeq B'$, if for any i_1, j_1, i_2, j_2, we have that $b_{i_1j_1} > b_{i_2j_2}$ implies $b'_{i_1j_1} \geq b'_{i_2j_2}$, and $b'_{i_1j_1} > b'_{i_2j_2}$ implies $b_{i_1j_1} \geq b_{i_2j_2}$. An *order consistent class* of bid profiles consists of bid profiles that are all pairwise order consistent. The set of all allowable bid profiles can be divided into a finite number of maximal order consistent classes (that is, order consistent classes that are not proper subsets of other order consistent classes). (Specifically, we have one such class for every strict ordering $<$ on the ordered pairs (i, j) ($1 \leq i \leq m$ and $1 \leq j \leq n$) such that $(i, j+1) < (i, j)$ and $(i+1, 1) < (i, 1)$ everywhere. We note that some bid profiles are part of more than one of these maximal order consistent classes: for example, the bid profile with all 0 elements belongs to all the classes.) We can apply the same definitions of order consistency and (maximal) order consistent classes to the profiles of *other* bids, the B_{-i}. Let $I(B_{-i})$ denote the maximal order consistent class that contains B_{-i}.[11]

[11] If B_{-i} belongs to multiple maximal order consistent classes, then $I(B_{-i})$ is the class with

The following definition of linear redistribution mechanisms successfully captures the ordering information of B_{-i}, by having separate coefficients for every maximal order consistent class.

$$r(B_{-i}) = c_{I(B_{-i}),0} + c_{I(B_{-i}),1} v_{-i,1} + \ldots + c_{I(B_{-i}),mn-m} v_{-i,mn-m}$$

Since $\begin{bmatrix} 0 & 0 \\ U & U \end{bmatrix}$ and $\begin{bmatrix} U & 0 \\ U & 0 \end{bmatrix}$ are not order consistent, they can result in different redistribution payments in this class of redistribution mechanisms.

Of course, this set of coefficients is unwieldy. As it turns out, we can simplify the representation of these mechanisms if we assume that they are continuous.

Let r be a linear redistribution mechanism (as just defined). Let $T(B_{-i}, k)$ be the result of changing the largest k elements of B_{-i} into U, and changing the remaining elements of B_{-i} into 0. (We assume that ties for the top k values are broken in a consistent way.) We note that $T(B_{-i}, k) \simeq B_{-i}$ for all $0 \leq k \leq mn-m$. For example, $T(\begin{bmatrix} 1 & 2 \\ 4 & 3 \end{bmatrix}, 1) = \begin{bmatrix} 0 & 0 \\ U & 0 \end{bmatrix}$ and $T(\begin{bmatrix} 1 & 2 \\ 4 & 3 \end{bmatrix}, 2) = \begin{bmatrix} 0 & 0 \\ U & U \end{bmatrix}$.

We define the following function r':

$$r'(B_{-i}) = r(T(B_{-i}, 0)) + \frac{r(T(B_{-i}, 1)) - r(T(B_{-i}, 0))}{U} v_{-i,1} + \ldots + \frac{r(T(B_{-i}, mn-m)) - r(T(B_{-i}, mn-m-1))}{U} v_{-i,mn-m}$$

Proposition 32. *If r is continuous, then $r = r'$.*

Proof. We first restrict our attention to profiles B_{-i} in a specific (but arbitrary) maximal order consistent class; moreover, we only consider profiles B_{-i} in which no two elements are equal. For any B_{-i} in this class, we use the same $mn - m + 1$

the smallest index in any predetermined order of all the classes. If we assume continuity of the redistribution function, as we will do below, then in fact it does not matter which maximal order consistent class we choose for B_{-i}.

coefficients of r, and $T(B_{-i}, k)$ (and hence $r(T(B_{-i}, k))$) depends only on k. That is, both the coefficients and $T(B_{-i}, k)$ are constant in B_{-i}.

If r is continuous, then when B_{-i} approaches $T(B_{-i}, k)$, we have that $r(B_{-i})$ approaches $r(T(B_{-i}, k))$. By the definition of r', we also have that when B_{-i} approaches $T(B_{-i}, k)$, that is, when the first k elements of B_{-i} approach U and the remaining elements of B_{-i} approach 0, we have that $r'(B_{-i})$ approaches $r(T(B_{-i}, k))$. That is, $r(T(B_{-i}, k)) = r'(T(B_{-i}, k))$ for $0 \le k \le mn - m$; that is, the functions agree in $mn-m+1$ different places. Since r and r' are both linear functions with $mn-m+1$ constant coefficients, r and r' must be the same function when B_{-i} is restricted to one class. Since the choice of class was arbitrary, we have that $r = r'$. □

From now on, we only consider continuous r. Hence, we can characterize r by the values it attains at all possible $T(B_{-i}, k)$. $T(B_{-i}, k)$ consists of only the numbers U and 0. We represent $T(B_{-i}, k)$ by an integer vector of length n, where the ith coordinate of the vector is the number of Us in the ith column of $T(B_{-i}, k)$.

For example,

$$T(\begin{bmatrix} 4 & 2 \\ 5 & 3 \end{bmatrix}, 2) = \begin{bmatrix} U & 0 \\ U & 0 \end{bmatrix} \rightarrow < 2, 0 >$$

$$T(\begin{bmatrix} 1 & 2 \\ 4 & 3 \end{bmatrix}, 3) = \begin{bmatrix} 0 & U \\ U & U \end{bmatrix} \rightarrow < 1, 2 >$$

Using this, $r(T(B_{-i}, k))$ can be rewritten as $r[x_1, x_2, \ldots, x_{n-1}]$, where $< x_1, x_2, \ldots, x_{n-1} >$ is the vector representing $T(B_{-i}, k)$ (with for each i, $0 \le x_i \le m$, and $\sum x_i = k$). Moreover, because we have, for example, that $r(\begin{bmatrix} 0 & U \\ U & U \end{bmatrix}) = r(\begin{bmatrix} U & 0 \\ U & U \end{bmatrix})$, we can assume without loss of generality that $x_1 \ge x_2 \ge \ldots \ge x_{n-1}$.

The following is an example of how to compute an agent's redistribution payment based on the values of $r[x_1, x_2, \ldots, x_{n-1}]$.

Example 6. Let $n = 3$ and $m = 2$. Let $B_{-i} = \begin{bmatrix} 3 & 1 \\ 4 & 2 \end{bmatrix}$.

$$r(B_{-i}) = r(T(B_{-i}, 0)) + \frac{r(T(B_{-i}, 1)) - r(T(B_{-i}, 0))}{U} v_{-i,1} + \ldots +$$

$$\frac{r(T(B_{-i}, mn - m)) - r(T(B_{-i}, mn - m - 1))}{U} v_{-i,mn-m}$$

$$= r(\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}) + \frac{r(\begin{bmatrix} 0 & 0 \\ U & 0 \end{bmatrix}) - r(\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix})}{U} \cdot 4 + \frac{r(\begin{bmatrix} U & 0 \\ U & 0 \end{bmatrix}) - r(\begin{bmatrix} 0 & 0 \\ U & 0 \end{bmatrix})}{U} \cdot 3$$

$$+ \frac{r(\begin{bmatrix} U & 0 \\ U & U \end{bmatrix}) - r(\begin{bmatrix} U & 0 \\ U & 0 \end{bmatrix})}{U} \cdot 2 + \frac{r(\begin{bmatrix} U & U \\ U & U \end{bmatrix}) - r(\begin{bmatrix} U & 0 \\ U & U \end{bmatrix})}{U} \cdot 1$$

$$= r[0, 0] + \frac{r[1, 0] - r[0, 0]}{U} \cdot 4 + \frac{r[2, 0] - r[1, 0]}{U} \cdot 3 + \frac{r[2, 1] - r[2, 0]}{U} \cdot 2 + \frac{r[2, 2] - r[2, 1]}{U} \cdot 1$$

Since the values of the $r[x_1, x_2, \ldots, x_{n-1}]$ completely characterize the continuous linear redistribution mechanism, we can solve for values of the $r[x_1, x_2, \ldots, x_{n-1}]$ for which the corresponding linear redistribution mechanism satisfies the non-deficit property and produces the least waste in expectation under this constraint.

The following proposition characterizes the non-deficit linear redistribution mechanisms.

Proposition 33. *A linear redistribution mechanism satisfies the non-deficit property if and only if the corresponding $r[x_1, x_2, \ldots, x_{n-1}]$ satisfy the following inequalities: For all $m \geq x_1 \geq x_2 \geq x_3 \geq \ldots \geq x_n \geq 0$, $\sum_{i=1}^{n} r[x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n] \leq U \cdot (\sum_{i=1}^{n} \min\{(\sum_{j=1}^{n} x_j) - x_i, m\} - (n-1) \min\{\sum_{j=1}^{n} x_j, m\})$. (The right-hand side of the inequality corresponds to the total VCG payment for the profile $< x_1, x_2, \ldots, x_n >$.)*

Proof. To see why the right-hand side $U \cdot (\sum_{i=1}^{n} \min\{(\sum_{j=1}^{n} x_j) - x_i, m\} - (n - 1) \min\{\sum_{j=1}^{n} x_j, m\})$ corresponds to the total VCG payment, we note that

$U \cdot \min\{(\sum_{j=1}^{n} x_j) - x_i, m\}$ is the total efficiency when i is removed, so that $U \cdot \sum_{i=1}^{n} \min\{(\sum_{j=1}^{n} x_j) - x_i, m\}$ is the sum of all the terms corresponding to efficiencies when one agent is removed. $U \cdot (n-1) \min\{\sum_{j=1}^{n} x_j, m\})$ corresponds to the sum of the basic Groves terms in the payments from the agents: in this term, each agent receives the total efficiency obtained by the other agents (when the agent is not removed), and if we sum over all the agents, that means each agent is counted $n-1$ times.

Now we can prove the main part of the proposition. If the non-deficit property is satisfied for all bid profiles, then it should also be satisfied when the marginal values are restricted to be either U or 0. This proves the "only if" direction. Now we prove the "if" direction. Let B be any bid profile from a fixed maximal order consistent class. This implies that the maximal order consistent class of B_{-i} is fixed as well, for every i. The total VCG payment equals the sum over all i of the m highest elements in B_{-i}, minus $n-1$ times the sum of the m highest elements in B. In either case, because we are restricting attention to a fixed class, the m highest elements are the same ones for any B in the class. Because of this, the VCG payments are linear in the v_i. Additionally, again because we are restricting attention to one particular class, the redistribution payments are also linear in the v_i.

Now, if the inequalities hold, that means that the total VCG payment minus the total redistribution is nonnegative when the marginal values are restricted to either U or 0. That is, the non-deficit constraints hold for these extreme cases. But by Lemma 3, if a non-deficit constraint is violated anywhere, then a non-deficit constraint must be violated for one of these extreme cases. It follows that the non-deficit constraints hold everywhere in the class that we were considering, and because this class was arbitrary, the non-deficit constraint must hold everywhere. □

Let z be the total number of maximal order consistent classes. Let Z_j be an arbitrary bid profile that is (only) in the jth class. Let $P(B \in I(Z^j))$ be the probability that a bid profile is drawn that is (only) in the jth class, and let $E(v_{-i,k}|B \in I(Z^j))$ be the expectation of the kth-highest marginal value among B_{-i}, given that B is (only) in the jth class. We assume that the probability that we draw a bid vector that is in more than one class is zero (this would require that two values are exactly equal).

Now we are ready to introduce a linear program that solves for the optimal-in-expectation linear redistribution mechanism.[12] This linear program is based on the alternative representation of linear redistribution mechanisms, whose correctness was established by Proposition 32, and on the characterization of the non-deficit constraints established for this representation by Proposition 33.

Variables: $r[x_1, x_2, \ldots, x_{n-1}]$ for all integer $m \geq x_1 \geq x_2 \geq \ldots \geq x_{n-1} \geq 0$
Maximize:

$$\sum_j P(B \in I(Z^j)) \sum_i [r(T(Z^j_{-i}, 0)) + \frac{r(T(Z^j_{-i},1)) - r(T(Z^j_{-i},0))}{U} E(v_{-i,1}|B \in I(Z^j)) + \ldots + \frac{r(T(Z^j_{-i},mn-m)) - r(T(Z^j_{-i},mn-m-1))}{U} E(v_{-i,mn-m}|B \in I(Z^j))]$$

Subject to:
For all $m \geq x_1 \geq x_2 \geq x_3 \geq \ldots \geq x_n \geq 0$,
$\sum_{i=1}^n r[x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n] \leq$
$U \cdot (\sum_{i=1}^n \min\{(\sum_{j=1}^n x_j) - x_i, m\} - (n-1) \min\{\sum_{j=1}^n x_j, m\})$

We do not have an analytical solution to this linear program; all that we can do is solve for the optimal mechanism for specific values of m and n. More problematically, in general it is not easy to compute the constants $P(B \in I(Z^j))$ and $E(v_{-i,k}|B \in I(Z^j))$. One way to work around this problem is to approximate the final

[12] Incidentally, we can give a similar linear program for finding the linear redistribution mechanism that is worst-case optimal, that is, it maximizes the fraction of total VCG payment redistributed in the worst case. In Section 2.1, we have already identified a worst-case optimal linear mechanism for the nonincreasing marginal values case; however, that mechanism is only optimal under the requirement of ex-post individual rationality. The linear programming technique here can be used to find the worst-case optimal mechanism when individual rationality is not required.

result. That is, instead of computing an exact optimal linear redistribution mechanism, we can draw a few sample bid profiles, and solve for a linear redistribution mechanism that is optimal for the samples. This way, we do not need to compute any probabilities or conditional expectations; we simply sum over the profiles in the sample in the objective. (However, we still enforce the constraints everywhere, not just on the samples.) Because the linear redistribution mechanisms are continuous and we assume continuous and bounded prior distributions for the valuations, it follows that as the number of samples grows, we approach an optimal mechanism.

We now return to the original idea for the definition of linear redistribution mechanisms: what if we ignore the ordering information and just use coefficients c_k for $0 \leq k \leq mn - m$, which do not depend on the maximal order consistent class? This will be a more scalable approach, although it will come at a loss. To find an optimal mechanism in this class, we can take a similar approach as we did above for the more general definition of linear redistribution mechanisms (and this approach is correct for similar reasons). We consider the extreme bid vectors where all marginal values are U or 0, represented by vectors of integers $x_1, x_2, \ldots, x_n$, as before. The fact that we ignore the ordering information now implies that we require that $r[x_1, x_2, \ldots, x_{n-1}] = r[y_1, y_2, \ldots, y_{n-1}]$ whenever $\sum_{i=1}^{n-1} x_i = \sum_{i=1}^{n-1} y_i$. So, we can rewrite $r[x_1, \ldots, x_{n-1}]$ as $r[\sum_{i=1}^{n-1} x_i]$. That is, the variables now are $r[k]$ for $k = 0, 1, \ldots, mn - m$. The redistribution function now becomes:

$$r(B_{-i}) = r[0] + \frac{r[1] - r[0]}{U} v_{-i,1} + \ldots + \frac{r[mn - m] - r[mn - m - 1]}{U} v_{-i,mn-m}$$

The linear program for finding an optimal mechanism becomes:

Variables: $r[k]$ for integer $0 \leq k \leq mn - m$
Maximize:

$\sum_i [r[0] + \frac{r[1]-r[0]}{U} E(v_{-i,1}) + \ldots + \frac{r[mn-m]-r[mn-m-1]}{U} E(v_{-i,mn-m})]$

Subject to:
For all $m \geq x_1 \geq x_2 \geq x_3 \geq \ldots \geq x_n \geq 0$,
$\sum_{i=1}^{n} r[(\sum_{j=1}^{n} x_j) - x_i] \leq$
$U \cdot (\sum_{i=1}^{n} \min\{(\sum_{j=1}^{n} x_j) - x_i, m\} - (n-1) \min\{\sum_{j=1}^{n} x_j, m\})$

While this linear program is much more manageable, it may lead to worse results than the earlier linear program, which optimizes over the more general class of linear redistribution mechanisms that take the ordering information into account. We now study some example solutions to this linear program, and compare them to the Bailey-Cavallo redistribution mechanism [8, 20]. We recall that the Bailey-Cavallo mechanism redistributes to every agent $\frac{1}{n}$ times the VCG payment that would result if this agent were removed from the auction. If we only consider bid profiles from a specific maximal order consistent class, then for any i, the VCG payment that would result if i is removed is a linear combination of the $v_{-i,k}$. Therefore, the Bailey-Cavallo mechanism belongs to the family of linear redistribution mechanisms that consider the ordering information (and hence, the optimal solution to the earlier linear program will do at least as well as the Bailey-Cavallo mechanism). The Bailey-Cavallo mechanism does not belong to the family of linear redistribution mechanisms that ignore the ordering information: in fact, we will see that it sometimes performs better than the optimal mechanism among linear redistribution mechanisms that ignore the ordering information. Hence, ignoring the ordering information in general comes at a cost.

For these examples, let us recall that agent i's bid vector B_i consists of m elements $b_{i1}, b_{i2}, \ldots, b_{im}$. In both examples, we assume that the values of $b_{i1}, b_{i2}, \ldots, b_{im}$ are drawn independently from the uniform $[0, 1]$ distribution, with b_{ij} being the jth highest among the m drawn values. We also assume that $B_1, B_2, \ldots, B_n$ are independent.

Example 7. Suppose that $n = 3$ and $m = 2$. By solving the above linear program (the one that ignores the ordering information), we get the following linear redistribution mechanism that ignores ordering information: $r(B_{-i}) = \frac{2}{3}v_{-i,3}$. That is, an agent's redistribution is equal to two thirds of the third highest marginal value among the set of other bids. The expected waste of this mechanism is 0.2571. In contrast, the expected waste of the Bailey-Cavallo mechanism is 0.4571. (The expected total VCG payment is 1.0571.) So, for this example, the optimal linear redistribution mechanism that ignores the ordering information outperforms the Bailey-Cavallo mechanism.

Example 8. Suppose that $n = 7$ and $m = 2$. By solving the above linear program (the one that ignores the ordering information), we get the following linear redistribution mechanism that ignores ordering information: $r(B_{-i}) = \frac{1}{5}v_{-i,3} + \frac{3}{35}v_{-i,4}$. That is, an agent's redistribution is equal to $\frac{1}{5}$ times the third highest marginal value among the set of other bids, plus $\frac{3}{35}$ times the fourth highest marginal value among the set of other bids. The expected waste of this mechanism is 0.0923. In contrast, the expected waste of the Bailey-Cavallo mechanism is 0.0671. (The expected total VCG payment is 1.5846.) So, for this example, the Bailey-Cavallo mechanism outperforms the optimal linear redistribution mechanism that ignores the ordering information.

In both of these examples (as well as in other examples for which we solved the linear program, including examples with other distributions), the optimal linear redistribution mechanism that ignores the ordering information is a special case of the following more general mechanism.

Mechanism M^* is defined as follows, where $t = m + \lfloor \frac{m(n-2)}{n} \rfloor$.

- $r[k] = U\frac{k-m}{n-2}$ for $k = m+1, m+2, \ldots, t$

- $r[k] = U\frac{m}{n}$ for $k > t$

The redistribution an agent receives is:

$$r(B_{-i}) = \sum_{m+1 \leq k \leq t} \frac{1}{n-2} v_{-i,k} + (\frac{m}{n} - \frac{t-m}{n-2}) v_{-i,t+1}$$

We conjecture that there are some more general conditions under which M^* is the optimal linear redistribution mechanism that ignores the ordering information.

2.3 Undominated VCG Redistribution Mechanisms

In the previous section, we mentioned a result (Proposition 21) that stated that OEL mechanisms are *collectively undominated*. In this section, we will prove this result and investigate the dominance notion in more detail. We study the problem of designing mechanisms whose redistribution functions are *undominated* in the sense that no other mechanisms can always perform as well, and sometimes better. (Here, "always" means for every profile of types.) We introduce two measures for comparing two VCG redistribution mechanisms with respect to how well off they make the agents. We say a non-deficit VCG redistribution mechanism is *individually undominated* if there exists no other non-deficit VCG redistribution mechanism that always has a larger or equal redistribution *for each agent*. We say a non-deficit VCG redistribution mechanism is *collectively undominated* if there exists no other non-deficit VCG redistribution mechanism that always has a larger or equal *sum of redistributions*. We study the question of finding maximal elements in the space of non-deficit redistribution mechanisms, with respect to the partial orders induced by both measures. For the first measure, we give a characterization of all individually undominated VCG redistribution mechanisms, and propose two techniques for generating individually undominated mechanisms based on known individually dominated mechanisms. Experimental results show that these techniques can significantly increase the agents' utilities. For the second measure, we characterize all collectively undominated VCG redistribution mechanisms that are anonymous and have linear payment functions, for auctions with multiple indistinguishable units, where each agent is only interested in a single copy of the unit.

2.3.1 Formalization

We still use n to denote the number of agents. We still use θ_i to denote agent i's true/reported type (we are restricting attention to strategy-proof mechanisms). Unless specified, we are dealing with general combinatorial auctions in this section.

A VCG redistribution mechanism is defined by a function $r_i : \Theta_1 \times \ldots \times \Theta_{i-1} \times \Theta_{i+1} \times \ldots \times \Theta_n \to \mathbb{R}$ for each agent i. That is, letting θ_{-i} be the vector of types submitted by agents other than i, $r_i(\theta_{-i})$ indicates the amount redistributed to i. For an anonymous redistribution mechanism, $r_i = r$ for all i.

Let us say that a VCG redistribution mechanism is *feasible* if it satisfies the non-deficit constraint.[13] The trivial redistribution mechanism that redistributes nothing is always feasible. As another example, Cavallo's mechanism [20] is given by $r_i(\theta_{-i}) = \frac{1}{n} \min_{\theta_i \in \Theta_i} VCG(\theta_i, \theta_{-i})$, where $VCG(\theta_i, \theta_{-i})$ is the total VCG payment collected for those reports. We could see that each agent's redistribution is at most $\frac{1}{n}$ of the total VCG payment, so that there is never a deficit.

2.3.2 Individual and Collective Dominance

How should we select a redistribution mechanism? In general, we prefer to redistribute as much as possible. However, two redistribution mechanisms may be incomparable in the sense that one redistributes more for one vector of reported types, and the other redistributes more for another vector. In this subsection, we define two measures for comparing two VCG redistribution mechanisms with respect to how well off they make the agents.

[13] It should be noted that here we do not require a feasible VCG redistribution mechanism to be individually rational. There are two reasons for this. First, as mentioned in Section 2.2, since our objective is to maximize social welfare, if the prior distribution over the agents' valuations is symmetric across agents, then under any redistribution mechanism that redistributes a nonnegative amount of payment in expectation, every agent benefits from participating in the mechanism (the agent receives nonnegative expected utility). That is, *ex-interim* individual rationality is not a binding constraint. Second, some of the results in this section are based on the OEL mechanisms, which ignore individual rationality. To be consistent, we also ignore individual rationality here.

Before discussing the above two measures, we first mathematically characterize all feasible VCG redistribution mechanisms.

Proposition 34. *A redistribution mechanism* $\mathbf{r} = (r_1, \ldots, r_n)$ *is* feasible *if and only if for all* i *and all* $\theta_1, \ldots, \theta_n$

$$r_i(\theta_{-i}) \leq \inf_{\theta_i' \in \Theta_i} \{VCG(\theta_i', \theta_{-i}) - \sum_{j \neq i} r_j(\theta_{-j}')\}^{14} \tag{2.1}$$

Here, θ_{-j}' *are the reported types of the agents other than* j *when* θ_i *is replaced by* θ_i'*.* $VCG(\theta_i', \theta_{-i})$ *is the total VCG payments for the type vector* $\theta_1, \ldots, \theta_{i-1}, \theta_i', \theta_{i+1}, \ldots, \theta_n$*.*

Proof. We first prove the "if" direction. For any i and $\theta_1, \ldots, \theta_n$, Equation 2.1 implies that $r_i(\theta_{-i}) \leq VCG(\theta_i', \theta_{-i}) - \sum_{j \neq i} r_j(\theta_{-j}')$ for any $\theta_i' \in \Theta_i$. If we let $\theta_i' = \theta_i$, we obtain $r_i(\theta_{-i}) + \sum_{j \neq i} r_j(\theta_{-j}) \leq VCG(\theta_i, \theta_{-i})$. Thus, the non-deficit property holds.

We now prove the "only if" direction. By the non-deficit property, for any i, any $\theta_1, \ldots, \theta_{i-1}, \theta_{i+1}, \ldots, \theta_n$, and any θ_i', we must have $r_i(\theta_{-i}) + \sum_{j \neq i} r_j(\theta_{-j}') \leq VCG(\theta_i', \theta_{-i})$, or equivalently $r_i(\theta_{-i}) \leq VCG(\theta_i', \theta_{-i}) - \sum_{j \neq i} r_j(\theta_{-j}')$. Since θ_i' is arbitrary, Equation 2.1 follows. □

Now we are ready to define individual and collective (un)dominance.

Definition 1. A redistribution mechanism $\mathbf{r}$ is *individually undominated* if it is feasible, and there does not exist a feasible redistribution mechanism $\mathbf{r}'$ that *individually dominates* it, that is,

- for all i, for all $\theta_1, \ldots, \theta_n$, $r_i'(\theta_{-i}) \geq r_i(\theta_{-i})$.
- for some i, for some $\theta_1, \ldots, \theta_n$, $r_i'(\theta_{-i}) > r_i(\theta_{-i})$.

[14] We use "inf" instead of "min", because $\mathbf{r}$ may not be continuous.

Definition 2. A redistribution mechanism $\mathbf{r}$ is *collectively undominated* if it is feasible, and there does not exist a feasible redistribution mechanism $\mathbf{r}'$ that *collectively dominates* it, that is,

- for all $\theta_1, \ldots, \theta_n$, $\sum_i r'_i(\theta_{-i}) \geq \sum_i r_i(\theta_{-i})$.
- for some $\theta_1, \ldots, \theta_n$, $\sum_i r'_i(\theta_{-i}) > \sum_i r_i(\theta_{-i})$.

It is easy to see that being collectively undominated is stronger than being individually undominated. Next, we give an example to show that being collectively undominated is strictly stronger.

Example 9. Consider a single-item auction with 4 players. We assume that for each player, the set of allowed types is the same, namely, integers from 0 to 3. Here, the VCG mechanism is just the second-price auction.

We define feasible redistribution mechanisms 1 and 2 as follows:

Mechanism 1: $r(\theta_{-i}) = r([\theta_{-i}]_1, [\theta_{-i}]_2, [\theta_{-i}]_3)$, and the function r is given in Table 2.5. ($[\theta_{-i}]_j$ is the jth highest type among types other than i's own type.)

Mechanism 2: $r'(\theta_{-i}) = r'([\theta_{-i}]_1, [\theta_{-i}]_2, [\theta_{-i}]_3)$, and the function r' is given in Table 2.5.

With the above characterization, mechanism 2 collectively dominates mechanism 1. The redistribution under mechanism 2 is never lower, and in some cases it is strictly higher: for example, for the type vector $(3, 2, 2, 2)$, the total redistribution under mechanism 1 is $1/2$, but the total redistribution under mechanism 2 is 1. On the other hand, mechanism 2 does not individually dominate mechanism 1: for example, $r(3, 3, 2) = 1 > 5/6 = r'(3, 3, 2)$. In fact, no feasible redistribution mechanism individually dominates mechanism 1.

Table 2.5: Example mechanisms for differentiating being collectively undominated and being individually undominated.

$\mathbf{r}(\mathbf{0},\mathbf{0},\mathbf{0})$	0	$\mathbf{r}'(\mathbf{0},\mathbf{0},\mathbf{0})$	0
$\mathbf{r}(\mathbf{1},\mathbf{0},\mathbf{0})$	0	$\mathbf{r}'(\mathbf{1},\mathbf{0},\mathbf{0})$	0
$\mathbf{r}(\mathbf{1},\mathbf{1},\mathbf{0})$	1/4	$\mathbf{r}'(\mathbf{1},\mathbf{1},\mathbf{0})$	1/4
$\mathbf{r}(\mathbf{1},\mathbf{1},\mathbf{1})$	1/4	$\mathbf{r}'(\mathbf{1},\mathbf{1},\mathbf{1})$	1/4
$\mathbf{r}(\mathbf{2},\mathbf{0},\mathbf{0})$	0	$\mathbf{r}'(\mathbf{2},\mathbf{0},\mathbf{0})$	0
$\mathbf{r}(\mathbf{2},\mathbf{1},\mathbf{0})$	1/12	$\mathbf{r}'(\mathbf{2},\mathbf{1},\mathbf{0})$	7/24
$\mathbf{r}(\mathbf{2},\mathbf{1},\mathbf{1})$	0	$\mathbf{r}'(\mathbf{2},\mathbf{1},\mathbf{1})$	1/6
$\mathbf{r}(\mathbf{2},\mathbf{2},\mathbf{0})$	1/2	$\mathbf{r}'(\mathbf{2},\mathbf{2},\mathbf{0})$	1/2
$\mathbf{r}(\mathbf{2},\mathbf{2},\mathbf{1})$	0	$\mathbf{r}'(\mathbf{2},\mathbf{2},\mathbf{1})$	1/4
$\mathbf{r}(\mathbf{2},\mathbf{2},\mathbf{2})$	1/2	$\mathbf{r}'(\mathbf{2},\mathbf{2},\mathbf{2})$	1/2
$\mathbf{r}(\mathbf{3},\mathbf{0},\mathbf{0})$	0	$\mathbf{r}'(\mathbf{3},\mathbf{0},\mathbf{0})$	0
$\mathbf{r}(\mathbf{3},\mathbf{1},\mathbf{0})$	1/4	$\mathbf{r}'(\mathbf{3},\mathbf{1},\mathbf{0})$	1/4
$\mathbf{r}(\mathbf{3},\mathbf{1},\mathbf{1})$	0	$\mathbf{r}'(\mathbf{3},\mathbf{1},\mathbf{1})$	1/4
$\mathbf{r}(\mathbf{3},\mathbf{2},\mathbf{0})$	2/3	$\mathbf{r}'(\mathbf{3},\mathbf{2},\mathbf{0})$	2/3
$\mathbf{r}(\mathbf{3},\mathbf{2},\mathbf{1})$	1	$\mathbf{r}'(\mathbf{3},\mathbf{2},\mathbf{1})$	19/24
$\mathbf{r}(\mathbf{3},\mathbf{2},\mathbf{2})$	0	$\mathbf{r}'(\mathbf{3},\mathbf{2},\mathbf{2})$	1/6
$\mathbf{r}(\mathbf{3},\mathbf{3},\mathbf{0})$	2/3	$\mathbf{r}'(\mathbf{3},\mathbf{3},\mathbf{0})$	5/6
$\mathbf{r}(\mathbf{3},\mathbf{3},\mathbf{1})$	0	$\mathbf{r}'(\mathbf{3},\mathbf{3},\mathbf{1})$	7/12
$\mathbf{r}(\mathbf{3},\mathbf{3},\mathbf{2})$	1	$\mathbf{r}'(\mathbf{3},\mathbf{3},\mathbf{2})$	5/6
$\mathbf{r}(\mathbf{3},\mathbf{3},\mathbf{3})$	0	$\mathbf{r}'(\mathbf{3},\mathbf{3},\mathbf{3})$	1/2

2.3.3 Individually Undominated Redistribution Mechanisms

In this subsection, we focus on individually undominated redistribution mechanisms. We first give a characterization of all individually undominated VCG redistribution mechanisms. Then, we propose two techniques for generating individually undominated mechanisms based on known individually dominated mechanisms. Finally, our experimental results show that these techniques can significantly increase the agents' utilities.

Theorem 7. *A redistribution mechanism* $\mathbf{r}$ *is individually undominated if and only if for all* i *and all* $\theta_1, \ldots, \theta_n$

$$r_i(\theta_{-i}) = \inf_{\theta'_i \in \Theta_i} \{VCG(\theta'_i, \theta_{-i}) - \sum_{j \neq i} r_j(\theta'_{-j})\} \tag{2.2}$$

Here, θ'_{-j} are the reported types of the agents other than j when θ_i is replaced by θ'_i.

It should be noted that the only difference between Equation 2.1 and Equation 2.2 is that "$\leq$" is replaced by "$=$".

Proof. We prove the "if" direction first. Any redistribution mechanism $\mathbf{r}$ that satisfies Equation 2.2 is feasible by Proposition 34. Now suppose that $\mathbf{r}$ is individually dominated, that is, there exists a feasible redistribution mechanism $\mathbf{r}'$ such that for all i and θ_{-i}, we have $r'_i(\theta_{-i}) \geq r_i(\theta_{-i})$, and for some i and θ_{-i}, we have $r'_i(\theta_{-i}) > r_i(\theta_{-i})$. For the i and θ_{-i} that make this inequality strict, we have $r'_i(\theta_{-i}) > r_i(\theta_{-i}) = \inf_{\theta'_i \in \Theta_i} \{VCG(\theta'_i, \theta_{-i}) - \sum_{j \neq i} r_j(\theta'_{-j})\} \geq \inf_{\theta'_i \in \Theta_i} \{VCG(\theta'_i, \theta_{-i}) - \sum_{j \neq i} r'_j(\theta'_{-j})\}$. But this contradicts the feasibility of $\mathbf{r}'$. It follows that $\mathbf{r}$ is individually undominated.

Now we prove the "only if" direction. Suppose Equation 2.2 is not satisfied. Then, there exists some i and θ_{-i} such that $r_i(\theta_{-i}) < \inf_{\theta'_i \in \Theta_i} \{VCG(\theta'_i, \theta_{-i}) - \sum_{j \neq i} r_j(\theta'_{-j})\}$. Let $a = \inf_{\theta'_i \in \Theta_i} \{VCG(\theta'_i, \theta_{-i}) - \sum_{j \neq i} r_j(\theta'_{-j})\} - r_i(\theta_{-i})$ (so that $a > 0$), and let $\mathbf{r}'$ be the same as $\mathbf{r}$, except that for the aforementioned i and θ_{-i}, $r'_i(\theta_{-i}) = r_i(\theta_{-i}) + a$. To show that this does not break the non-deficit constraint, consider any type vector (θ_i, θ_{-i}) where i and θ_{-i} are the same as before (that is, any type vector that is affected). Then, $r'_i(\theta_{-i}) = a + r_i(\theta_{-i}) = \inf_{\theta'_i \in \Theta_i} \{VCG(\theta'_i, \theta_{-i}) - \sum_{j \neq i} r_j(\theta'_{-j})\} = \inf_{\theta'_i \in \Theta_i} \{VCG(\theta'_i, \theta_{-i}) - \sum_{j \neq i} r'_j(\theta'_{-j})\}$. Thus, by Proposition 34, $\mathbf{r}'$ is feasible. This contradicts that $\mathbf{r}$ is individually undominated. Hence, Equation 2.2 must hold. $\square$

As an aside, suppose we were only interested in anonymous mechanisms, and we would therefore only consider a mechanism individually dominated if it were individually dominated by an *anonymous* mechanism. Then, the characterization in

Theorem 7 remains identical.[15] Therefore, all of our results apply to this modified definition as well.

An individually undominated redistribution mechanism always exists; in general, it is not unique. We now give two examples of individually undominated redistribution mechanisms.

Example 10. Consider a single-item auction with $n \geq 3$ agents. Agent i bids $\theta_i \in [0, \infty)$. Let $p(j, \theta)$ be the jth highest element of θ. If $\mathbf{r}$ is Cavallo's mechanism, then $r(\theta_{-i}) = \frac{1}{n}p(2, \theta_{-i})$ (Cavallo's mechanism is anonymous, so we omit the subscript of r.) To show r is individually undominated, it suffices to show Equation 2.2 is satisfied. For Equation 2.2, we first observe that for all θ'_i, $VCG(\theta'_i, \theta_{-i}) = p(2, (\theta'_i, \theta_{-i})) \geq p(2, \theta_{-i})$ and for all $j \neq i$, $VCG(\theta'_i, \theta_{-i}) = p(2, (\theta'_i, \theta_{-i})) \geq p(2, \theta'_{-j})$. Because $r_i(\theta_{-i}) + \sum_{j \neq i} r_j(\theta'_{-j}) = \frac{1}{n}p(2, \theta_{-i}) + \frac{1}{n} \sum_{j \neq i} p(2, \theta'_{-j})$, it follows that for all θ'_i, $r_i(\theta_{-i}) \leq VCG(\theta'_i, \theta_{-i}) - \sum_{j \neq i} r_j(\theta'_{-j})$. Moreover, if $\theta'_i = p(2, \theta_{-i})$, then all of the above inequalities become equalities. Hence Equation 2.2 holds. It follows that Cavallo's mechanism is individually undominated in this setting. (We will show that it is not individually undominated in more general settings.)

Example 11. Consider again a single-item auction with $n \geq 5$ agents. Agent i bids θ_i. Let $\mathbf{r}$ be the following anonymous redistribution mechanism: $r(\theta_{-i}) = \frac{1}{n-2}p(2, \theta_{-i}) - \frac{2}{(n-2)(n-3)}p(3, \theta_{-i}) + \frac{6}{n(n-2)(n-3)}p(4, \theta_{-i})$. Equation 2.2 can be shown to hold (the equality in Equation 2.2 is achieved by setting $\theta'_i = p(4, \theta_{-i})$).

Because in general, there are multiple individually undominated redistribution mechanisms, it is not clear which one is the best. If a prior distribution over agents' types is available, then we would prefer the one that redistributes the most in expectation; however, in this section, we do not wish to assume that such a prior is

[15] This can be proved by modifying the proof of Theorem 7, adding a/n to each agent's redistribution function instead of adding a to one agent's redistribution function.

available. Nevertheless, for any (feasible) redistribution mechanism that we might consider using, if it is individually dominated, then there exists another (feasible) redistribution mechanism that always redistributes at least as much to each agent, and more in some cases. Thus, in expectation, the latter mechanism redistributes at least as much for any prior distribution, and strictly more if the prior assigns positive probability to the set of type vectors on which the latter mechanism redistributes more. Hence, we would certainly prefer the latter mechanism—and if that mechanism is not individually undominated, we would prefer to find one that individually dominates it, *etc.* But how do we find such an improved mechanism? This is what we study next.

In what follows, we propose two techniques that, given a redistribution mechanism that is feasible and individually dominated, find a feasible redistribution mechanism that individually dominates it. (If the initial mechanism is already individually undominated, then the techniques will return the same mechanism.) One technique immediately produces an individually undominated mechanism that is not anonymous; the other techniques preserve anonymity, and after repeated application converge to an individually undominated mechanism. We emphasize that we can start with *any* feasible redistribution mechanism, including Cavallo's mechanism, the WCO mechanism (which, even though is optimal in the worst case, is generally not individually undominated), or even the trivial redistribution mechanism that redistributes nothing. These techniques can also be useful in settings where we do have a prior distribution. For example, after designing a redistribution mechanism based on a prior distribution, we can further improve it and make it individually undominated, which will never decrease the redistribution payment to any agent.

A Priority-Based Technique

Given a feasible redistribution mechanism $\mathbf{r}$ and a priority order over agents π, we can improve $\mathbf{r}$ into an individually undominated redistribution mechanism that is not anonymous. The technique works as follows.

1) Let $\pi : \{1, \ldots, n\} \to \{1, \ldots, n\}$ be a permutation representing the priority order. That is, $\pi(i)$ is agent i's priority value (the lower the value, the higher the priority). $\pi^{-1}(k)$ is the agent with the kth-highest priority.

2) Let $i = \pi^{-1}(1)$, and update i's redistribution function to $r_i^\pi(\theta_{-i}) = \inf_{\theta_i' \in \Theta_i} \{ VCG(\theta_i', \theta_{-i}) - \sum_{\pi(j)>1} r_j(\theta_{-j}')\}$. That is, we redistribute as much as possible to this agent without breaking feasibility.

3) We will now consider the remaining agents in turn, according to the order π. In the kth step, we update the redistribution function of agent $i = \pi^{-1}(k)$ to $r_i^\pi(\theta_{-i}) = \inf_{\theta_i' \in \Theta_i} \{VCG(\theta_i', \theta_{-i}) - \sum_{\pi(j)>k} r_j(\theta_{-j}') - \sum_{\pi(j)<k} r_j^\pi(\theta_{-j}')\}$. That is, we redistribute as much as possible to this agent without breaking feasibility, taking the previous $k-1$ updates into account.

Thus, for every agent i, $r_i^\pi(\theta_{-i}) = \inf_{\theta_i' \in \Theta_i} \{VCG(\theta_i', \theta_{-i}) - \sum_{\pi(j)>\pi(i)} r_j(\theta_{-j}') - \sum_{\pi(j)<\pi(i)} r_j^\pi(\theta_{-j}')\}$. The new redistribution mechanism $\mathbf{r}^\pi$ satisfies the following properties:

Proposition 35. *For all i, for all θ_{-i}, $r_i^\pi(\theta_{-i}) \geq r_i(\theta_{-i})$.*

Proof. First consider $i = \pi^{-1}(1)$, the agent with the highest priority. For any θ_{-i}, we have $r_i^\pi(\theta_{-i}) = \inf_{\theta_i' \in \Theta_i} \{VCG(\theta_i', \theta_{-i}) - \sum_{j \neq i} r_j(\theta_{-j}')\}$. Since the original redistribution mechanism $\mathbf{r}$ is feasible, by Equation 2.1, we have $r_i(\theta_{-i}) \leq \inf_{\theta_i' \in \Theta_i} \{VCG(\theta_i', \theta_{-i}) -$

$\sum_{j \neq i} r_j(\theta'_{-j})\}$. Hence $r_i^\pi(\theta_{-i}) \geq r_i(\theta_{-i})$.

For any $i \neq \pi^{-1}(1)$, $r_i^\pi(\theta_{-i}) = r_i(\theta_{-i}) + \inf_{\theta'_i \in \Theta_i}\{VCG(\theta'_i, \theta_{-i}) - r_i(\theta_{-i}) - \sum_{\pi(j)>\pi(i)} r_j(\theta'_{-j}) - \sum_{\pi(j)<\pi(i)} r_j^\pi(\theta'_{-j})\}$. We must show $\inf_{\theta'_i \in \Theta_i}\{VCG(\theta'_i, \theta_{-i}) - r_i(\theta_{-i}) - \sum_{\pi(j)>\pi(i)} r_j(\theta'_{-j}) - \sum_{\pi(j)<\pi(i)} r_j^\pi(\theta'_{-j})\} \geq 0$.

Consider $p = \pi^{-1}(\pi(i) - 1)$ (the agent immediately before i in terms of priority). For any θ_i, θ_{-i}, we have $VCG(\theta_i, \theta_{-i}) - r_i(\theta_{-i}) - \sum_{\pi(j)>\pi(i)} r_j(\theta_{-j}) - \sum_{\pi(j)<\pi(i)} r_j^\pi(\theta_{-j}) = VCG(\theta_i, \theta_{-i}) - \sum_{\pi(j)>\pi(p)} r_j(\theta_{-j}) - \sum_{\pi(j)<\pi(p)} r_j^\pi(\theta_{-j}) - r_p^\pi(\theta_{-p}) \geq \inf_{\theta'_p \in \Theta_p}\{VCG(\theta'_p, \theta_{-p}) - \sum_{\pi(j)>\pi(p)} r_j(\theta'_{-j}) - \sum_{\pi(j)<\pi(p)} r_j^\pi(\theta'_{-j})\} - r_p^\pi(\theta_{-p}) = 0$. (For the above inequality only, θ'_{-j} is the set of types reported by the agents other than j when θ_p is replaced by θ'_p.) Because θ_i is arbitrary, it follows that $\inf_{\theta'_i \in \Theta_i}\{VCG(\theta'_i, \theta_{-i}) - r_i(\theta_{-i}) - \sum_{\pi(j)>\pi(i)} r_j(\theta'_{-j}) - \sum_{\pi(j)<\pi(i)} r_j^\pi(\theta'_{-j})\} \geq 0$. It follows that $r_i^\pi(\theta_{-i}) \geq r_i(\theta_{-i})$ for all i and θ_{-i}. □

Proposition 36. $\mathbf{r}^\pi$ *is an individually undominated redistribution mechanism.*

Proof. Let $i = \pi^{-1}(n)$. For all $\theta_1, \ldots, \theta_n$, the total VCG payment that is not redistributed by $\mathbf{r}^\pi$ is $VCG(\theta_1, \ldots, \theta_n) - \sum_{j=1,\ldots,n} r_j^\pi(\theta_{-j}) \geq \inf_{\theta'_i \in \Theta_i}\{VCG(\theta'_i, \theta_{-i}) - \sum_{j \neq i} r_j^\pi(\theta'_{-j})\} - r_i^\pi(\theta_{-i}) = 0$. Hence $\mathbf{r}^\pi$ never incurs a deficit. So, $\mathbf{r}^\pi$ is feasible.

Using Proposition 35, we have $r_i^\pi(\theta_{-i}) = \inf_{\theta'_i \in \Theta_i}\{VCG(\theta'_i, \theta_{-i}) - \sum_{\pi(j)>\pi(i)} r_j(\theta'_{-j}) - \sum_{\pi(j)<\pi(i)} r_j^\pi(\theta'_{-j})\} \geq \inf_{\theta'_i \in \Theta_i}\{VCG(\theta'_i, \theta_{-i}) - \sum_{j \neq n} r_j^\pi(\theta'_{-j})\}$. Because $\mathbf{r}^\pi$ is feasible, the opposite inequality must also be satisfied (Equation 2.1)—hence we must have equality, that is, Equation 2.2 must hold. It follows that $\mathbf{r}^\pi$ is individually undominated. □

Example 12. Consider a single-item auction with four agents $1, 2, 3, 4$. In this setting, the redistribution under the WCO mechanism to agent i is $r(\theta_{-i}) = (2/7)p(2, \theta_{-i}) -$

$(1/7)p(3,\theta_{-i})$ (where $p(k,\theta_{-i})$ is the kth highest bid among bids other than i's). Consider a specific set of bids $(8,10,13,5)$ and let $\pi(i) = i$ for all i. (That is, agent 1 bids 8 for the item and has the highest priority, *etc.*) If we apply the above technique, the resulting redistribution payment to agent 1 is $r_1^\pi(10,13,5) = \inf_{\theta_1' \in [0,\infty)} \{VCG(\theta_1',10,13,5) - r(\theta_1',13,5) - r(\theta_1',10,5) - r(\theta_1',10,13)\}$ (where r is the WCO mechanism). It turns out that the expression is minimized at $\theta_1' = 0$, so that $r_1^\pi(10,13,5) = \frac{30}{7}$. This is twice the amount 1 would have received under WCO: $r(10,13,5) = (2/7)\cdot 10 - (1/7)\cdot 5 = \frac{15}{7}$.

For agent 2, $r_2^\pi(8,13,5) = \inf_{\theta_2' \in [0,\infty)} \{VCG(8,\theta_2',13,5) - r_1^\pi(\theta_2',13,5) - r(8,\theta_2',5) - r(8,\theta_2',13)\}$. This expression is minimized at $\theta_2' = 8$, so that $r_2^\pi(8,13,5) = \frac{17}{7}$. (Under WCO, 2 receives only $\frac{11}{7}$.)

For agent 3, $r_3^\pi(8,10,5) = \inf_{\theta_3' \in [0,\infty)} \{VCG(8,10,\theta_3',5) - r_1^\pi(10,\theta_3',5) - r_2^\pi(8,\theta_3',5) - r(8,10,\theta_3')\}$. This expression is minimized at $\theta_3' = 8$, so that $r_3^\pi(8,10,5) = \frac{11}{7}$. (Under WCO, 3 receives $\frac{11}{7}$ as well.)

For agent 4 $r_4^\pi(8,10,13) = \inf_{\theta_4' \in [0,\infty)} \{VCG(8,10,13,\theta_4') - r_1^\pi(10,13,\theta_4') - r_2^\pi(8,13,\theta_4') - r_3^\pi(8,10,\theta_4')\}$. This expression is minimized at $\theta_4' = 5$, so that $r_4^\pi(8,10,13) = \frac{12}{7}$. (Under WCO, 4 receives $\frac{12}{7}$ as well.)

We note that for this priority order, the total amount redistributed is $\frac{30+17+11+12}{7} = 10$, that is, all of the VCG payments are redistributed. This is not true for all priority orders; averaging over all priority orders, 0.315 remains unredistributed (compared to 3 for the WCO mechanism). Table 2.6 shows the results for all priority orders for this example.

Generally, most of the increase in redistribution payment goes to high-priority agents. Hence, a reasonable approximation can be obtained by only updating the redistribution payment functions of the first few agents. This still results in a feasible mechanism that individually dominates the original (or is the same), but it is no longer guaranteed to be individually undominated.

Iterative Techniques that Preserve Anonymity

The earlier technique will, in general, not produce an anonymous redistribution mechanism, even if the original mechanism $\mathbf{r}$ is anonymous. This is because agents higher in the priority order tend to receive higher redistribution payments. Here, we will introduce techniques that preserve anonymity.

One way to obtain an anonymous mechanism is to consider r^π for *all* permutations π, and take the average. That is, let $\bar{\mathbf{r}}$ be defined by $\bar{r}_i = \frac{1}{n!} \sum_{\pi \in S_n} (r_i^\pi)$, where S_n is the set of all permutations of n elements. Given that the setting and the initial mechanism are anonymous, this results in an anonymous mechanism. It is also feasible:

Proposition 37. *Any convex combination of a set $\{\mathbf{r^{(1)}}, \ldots, \mathbf{r^{(t)}}\}$ of feasible redistribution mechanisms is itself feasible.*

Proof. Let $\sum_{k=1}^{t} \alpha_k = 1$ with each $\alpha_k \geq 0$; we must show that $\mathbf{r} = \sum_{k=1}^{t} \alpha_k \mathbf{r^{(k)}}$ is feasible. For any i and θ_{-i}, for any k, we have $r_i^{(k)}(\theta_{-i}) \geq 0$, hence $r_i(\theta_{-i}) = \sum_{k=1}^{t} \alpha_k r_i^{(k)}(\theta_{-i}) \geq 0$. This implies individual rationality. Also, for any $\theta_1, \ldots, \theta_n$, for any k, $\sum_{i=1}^{n} r_i^{(k)}(\theta_{-i}) \leq VCG(\theta_1, \ldots, \theta_n)$, hence $\sum_{i=1}^{n} r_i(\theta_{-i}) = \sum_{k=1}^{t} \alpha_k \sum_{i=1}^{n} r_i^{(k)}(\theta_{-i}) \leq VCG(\theta_1, \ldots, \theta_n)$. This implies the non-deficit property. □

Because $\bar{\mathbf{r}}$ is anonymous, all $\bar{r}_i$ are the same, so we will simply use $\bar{r}$. Even though $\bar{r}$ is an average of a set of individually undominated redistribution mechanisms, in general, it itself is not individually undominated. In principle, we can take the resulting mechanism and apply the technique again. Unfortunately, this approach is not computationally practical—in fact, it may not be feasible to perform even one iteration of this technique if n is large, since we have to take an average over $n!$ mechanisms.[16] However, as we mentioned, it is also possible to apply the priority-based technique only to the first h agents. This still results in a feasible (but not necessarily individually undominated) mechanism, and tends to obtain most of the increase in redistribution payments. Taking the average over all such mechanisms is feasible for sufficiently small h (there will be $P_h^n = n!/(n-h)!$ such mechanisms), and will result in an anonymous mechanism. We will consider the extreme case where $h = 1$ (*i.e.* we only change one agent's redistribution function), so that we have to take an average over only n mechanisms. This we can do iteratively.

Given a feasible and anonymous redistribution mechanism r, let $r^0 = r$, and let r^k be the mechanism that results after k iterations of the above technique (with $h = 1$). Then, for all i and $\theta_1, \ldots, \theta_n$, $r^{k+1}(\theta_{-i}) = \frac{n-1}{n} r^k(\theta_{-i}) + \frac{1}{n} \inf_{\theta_i' \in \Theta_i} \{VCG(\theta_i', \theta_{-i}) - \sum_{j \neq i} r^k(\theta_{-j}')\}$.

This technique can be interpreted as a generalization of the basic idea underlying Cavallo's mechanism. We can rewrite $r^{k+1}(\theta_{-i}) = r^k(\theta_{-i}) + \frac{1}{n} \inf_{\theta_i' \in \Theta_i} \{VCG(\theta_i', \theta_{-i}) - \sum_{j \neq i} r^k(\theta_{-j}') - r^k(\theta_{-i})\}$. If the starting mechanism $r = r^0$ is the trivial redistribution mechanism that redistributes nothing, then $r^1(\theta_{-i}) = \frac{1}{n} \inf_{\theta_i' \in \Theta_i} \{VCG(\theta_i', \theta_{-i})\}$, which is exactly Cavallo's mechanism.

Proposition 38. *If r^k is feasible, r^{k+1} is feasible.*

---

[16] Computational limitations often prevent us from using certain mechanisms. As an extreme example, it is possible to have a computer search over the space of all possible (truthful) mechanisms for the setting at hand and find the best one [30] (more details in Section 1.2), but this does not scale to very large instances. By contrast, here, we have an analytical characterization of the mechanism, but computing its outcomes is still hard.

Proof. r^{k+1} is an average of feasible mechanisms, so Proposition 37 applies. □

Proposition 39. *For any i and θ_{-i}, $r^k(\theta_{-i})$ is nondecreasing in k.*

Proof. $r^{k+1}(\theta_{-i}) = r^k(\theta_{-i}) + \frac{1}{n} \inf_{\theta'_i \in \Theta_i} \{VCG(\theta'_i, \theta_{-i}) - \sum_{j \neq i} r^k(\theta'_{-j}) - r^k(\theta_{-i})\}$. Because r^k is feasible by Proposition 38, $\inf_{\theta'_i \in \Theta_i} \{VCG(\theta'_i, \theta_{-i}) - \sum_{j \neq i} r^k(\theta'_{-j}) - r^k(\theta_{-i})\} \geq 0$. Hence $r^{k+1}(\theta_{-i}) \geq r^k(\theta_{-i})$. □

Proposition 40. *As $k \to \infty$, r^k converges (point-wise) to an individually undominated redistribution mechanism.*

Proof. By Proposition 39, the $r^k(\theta_{-i})$ are nondecreasing in k, and since every r^k is feasible by Proposition 38, they must be bounded; hence they must converge (point-wise). For any i and θ_{-i}, let $d_k = \inf_{\theta'_i \in \Theta_i} \{VCG(\theta'_i, \theta_{-i}) - \sum_{j \neq i} r^k(\theta'_{-j})\} - r^k(\theta_{-i})$. Using Proposition 39, we derive the following inequality: $d_{k+1} = \inf_{\theta'_i \in \Theta_i} \{VCG(\theta'_i, \theta_{-i}) - \sum_{j \neq i} r^{k+1}(\theta'_{-j})\} - r^{k+1}(\theta_{-i}) \leq \inf_{\theta'_i \in \Theta_i} \{VCG(\theta'_i, \theta_{-i}) - \sum_{j \neq i} r^k(\theta'_{-j})\} - r^{k+1}(\theta_{-i}) = \inf_{\theta'_i \in \Theta_i} \{ VCG(\theta'_i, \theta_{-i}) - \sum_{j \neq i} r^k(\theta'_{-j})\} - \frac{n-1}{n} r^k(\theta_{-i}) - \frac{1}{n} \inf_{\theta'_i \in \Theta_i} \{VCG(\theta'_i, \theta_{-i}) - \sum_{j \neq i} r^k(\theta'_{-j})\} = \frac{n-1}{n} \inf_{\theta'_i \in \Theta_i} \{VCG(\theta'_i, \theta_{-i}) - \sum_{j \neq i} r^k(\theta'_{-j})\} - \frac{n-1}{n} r^k(\theta_{-i}) = \frac{n-1}{n} d_k$. As $k \to \infty$, $d_k = \inf_{\theta'_i \in \Theta_i} \{ VCG(\theta'_i, \theta_{-i}) - \sum_{j \neq i} r^k(\theta'_{-j})\} - r^k(\theta_{-i}) \to 0$. So in the limit, Equation 2.2 is satisfied. Thus, r^k converges (point-wise) to an individually undominated redistribution mechanism. □

Corollary 4. *If $r^{k+1} = r^k$, then r^k is individually undominated.*

Proposition 41. *If r^k is not individually undominated, then r^{k+1} individually dominates r^k.*

Proof. r^{k+1} always redistributes at least as much as r^k to each agent by Proposition 39. Moreover, $r^{k+1} \neq r^k$ (otherwise Corollary 4 would imply r^k is individually undominated). Hence there must be a case where r^{k+1} redistributes more than r^k. □

Next, we present the results of some experiments in which we use the techniques we proposed to improve both the WCO mechanism and Cavallo's mechanism. For the purpose of completeness, we also apply the nonanonymous (priority-based) technique to the trivial redistribution mechanism that redistributes nothing, and compare the resulting mechanism's performance with that of Cavallo's mechanism. (We omit the result of applying the anonymity-preserving technique to the trivial redistribution mechanism because, as we mentioned, after one iteration, we just obtain Cavallo's mechanism. We also omit the result of applying the nonanonymous technique to the trivial redistribution mechanism in multi-unit auctions with unit demand, because the resulting mechanism always has the same expected redistribution amount as Cavallo's mechanism: $m(m+1)/n$ times the $m+2$th highest bid, plus $m(n-m-1)/n$ times the $m+1$th highest bid.)

Improving the WCO mechanism. The WCO mechanism applies only to multi-unit auctions with unit demand (*i.e.* in which each agent only wants a single unit); in this setting, this mechanism maximizes the percentage that is redistributed in the worst case. This, however, does not mean that it is individually undominated, because it could be individually dominated by another mechanism that does equally well in the worst case, and better in other cases. Indeed, we can improve the WCO mechanism using our techniques (resulting in another, better, worst-case optimal mechanism).

For various m (number of units) and n (number of agents), we generated 100 random instances with each agent's valuation drawn uniformly from $[0, 1]$. Table 2.7 shows the ratio between the average amount that is not redistributed by the new mechanism (which results from applying one of our techniques to the WCO mechanism), and the average amount that is not redistributed by the (original) WCO mechanism. That is, it is the percentage of the amount that WCO fails to redistribute that the new mechanism also fails to redistribute. Lower numbers are better—100% indicates no improvement over WCO, 0% indicates that everything is redistributed. For the nonanonymous (priority-based) technique, to save computation time, we only update the redistribution payments for the first three agents. This technique redistributes more than the anonymity-preserving technique.

Improving Cavallo's mechanism. We recall that Cavallo's mechanism is individually undominated in the single-item auction setting (in fact, this remains true for multi-unit auctions with unit demand). However, as the experiment below shows, it is not individually undominated in general.

For a combinatorial auction with n single-minded agents and 2 items, we generated 100 random instances. For each agent, we randomly chose a nonempty bundle of items, and randomly chose a per-item value from $[0, 1]$ (which is multiplied by two if the agent desires the bundle of two items). The percentages have the same meaning as before. We distinguish between the known single-minded case (where the auctioneer knows which bundle the agent wants) and the unknown case. Again, the nonanonymous technique redistributes more; also, more is redistributed in the known case. The experimental results are shown in Table 2.8.

For the same set of 100 random instances, Table 2.9 shows the ratio between the average amount that is not redistributed by the mechanism which results from applying the nonanonymous technique to the trivial redistribution mechanism, and the average amount that is not redistributed by Cavallo's mechanism.

2.3.4 Collectively Undominated Redistribution Mechanisms

In this subsection, we characterize all collectively undominated VCG redistribution mechanisms that are anonymous and have linear payment functions, for auctions with multiple indistinguishable units, where each agent is only interested in a single copy of the unit.

First of all, we show that for multi-unit auctions with unit demand, the OEL mechanisms, which are anonymous and have linear payment functions, are collectively undominated (Proposition 21).

Proof. We only need to prove that the OEL mechanisms are not collectively dominated by any anonymous feasible redistribution mechanism. Suppose a nonanonymous feasible redistribution mechanism r collectively dominates an OEL mechanism r', then by permuting the indices of the agents under both mechanisms, the permuted r still collectively dominates the permuted r'. Since r' is anonymous, the permuted r' is just r'. That is, any permuted r collectively dominates r'. Now we take the average of all permuted r corresponding to all possible permutations of the agents. The resulting mechanism $\bar{r}$ is anonymous, and it collectively dominates r'. Therefore, if a nonanonymous feasible redistribution mechanism collectively dominates an OEL mechanism, then there exists an anonymous feasible redistribution mechanism that collectively dominates an OEL mechanism. Therefore, we only need to prove that the OEL mechanisms are not collectively dominated by any other anonymous feasible redistribution mechanisms.

Since we are dealing with multi-unit auctions with unit demand, an agent's type is just a single value. We use $[\theta]_i$ to denote the ith highest type among the agents. We still use U and L to denote the upper bound and the lower bound on the agents' types, respectively.

We recall that an OEL mechanism corresponds to an index from 0 to n. For

$k = 1, 2, \ldots, n-1$, the waste under the OEL mechanism with index k equals $m\binom{n-1}{m}/\binom{n}{k}([\theta]_k - [\theta]_{k+1})$. The waste under the OEL mechanism with index 0 equals $m\binom{n-1}{m}(U - [\theta]_1)$. The waste under the OEL mechanism with index n equals $m\binom{n-1}{m}([\theta]_n - L)$.

We first prove that for $k \in \{1, \ldots, n-1\}$, the OEL mechanism with index k is collectively undominated.

Suppose a feasible anonymous redistribution mechanism (corresponding to the redistribution function) r collectively dominates an OEL mechanism (corresponding to the redistribution function) r' with index $k \in \{1, \ldots, n-1\}$.

For any i and θ_{-i}, we define the following function: $\Delta(\theta_{-i}) = r(\theta_{-i}) - r'(\theta_{-i})$.

Since r collectively dominates r', we have that for any type profile, $\sum_{i=1}^{n} \Delta(\theta_{-i}) \geq 0$.

We also have that, whenever $[\theta]_{k+1} = [\theta]_k$, there is no waste under r'. In this case, because r is feasible, its total redistribution must equal the total redistribution under r' (otherwise, there will be deficit). Hence, whenever $[\theta]_{k+1} = [\theta]_k$, we have $\sum_{i=1}^{n} \Delta(\theta_{-i}) = 0$.

Now we claim that $\Delta(\theta_{-i}) = 0$ for all θ_{-i}.

Let $c(\theta_{-i})$ be the number of types among θ_{-i} that equal $[\theta_{-i}]_k$ (the kth highest type among θ_{-i}). Hence, we must show that for all θ_{-i} with $c(\theta_{-i}) \geq 1$, we have $\Delta(\theta_{-i}) = 0$.

We now prove it by induction on the value of c (backwards, from $n-1$ to 1).

Base case: $c = n - 1$.

Suppose there is a θ_{-i} with $c(\theta_{-i}) = n - 1$. That is, all the types in θ_{-i} are identical. When θ_i is also equal to the types in θ_{-i}, all types are the same so that $[\theta]_{k+1} = [\theta]_k$. Hence, by our earlier observation, we have $\sum_{j=1}^{n} \Delta(\theta_{-j}) = 0$. But we know that for all j, $\Delta(\theta_{-j})$ is the same value. Hence $\Delta(\theta_{-i}) = 0$ for all θ_{-i} when

$c(\theta_{-i}) = n - 1$.

Induction step.

Let us assume that for all θ_{-i}, if $c(\theta_{-i}) \geq p$ (where $p \in \{2, \ldots, n-1\}$), then $\Delta(\theta_{-i}) = 0$. Now we consider any θ_{-i} with $c(\theta_{-i}) = p - 1$. When θ_i is equal to $[\theta_{-i}]_k$, we have $[\theta]_k = [\theta]_{k+1}$, which implies that $\sum_{j=1}^{n} \Delta(\theta_{-j}) = 0$. For all j with $\theta_j = [\theta_{-i}]_k$, $\Delta(\theta_{-j}) = \Delta(\theta_{-i})$, and for other j, $c(\theta_{-j}) = p$. Therefore, by the induction assumption, $\sum_{j=1}^{n} \Delta(\theta_{-j})$ is a positive multiple of $\Delta(\theta_{-i})$, which implies that $\Delta(\theta_{-i}) = 0$.

By induction, we have shown that $\Delta(\theta_{-i}) = 0$ for all θ_{-i}. This implies that r and r' are identical. Hence, no other feasible anonymous redistribution mechanism collectively dominates an OEL mechanism with index $k \in \{1, \ldots, n-1\}$.

Similarly, we can prove that no other feasible anonymous redistribution mechanism collectively dominates an OEL mechanism with index $k = 0$ or $k = n$. (In the above induction steps, we set θ_i to be equal to $[\theta_{-i}]_k$. To prove that no other feasible anonymous redistribution mechanism collectively dominates an OEL mechanism with index $k = 0$, we use the same induction steps, except that we set θ_i to be equal to U. To prove that no other feasible anonymous redistribution mechanism collectively dominates an OEL mechanism with index $k = n$, we use the same induction steps, except that we set θ_i to be equal to L.) □

Next, we show that for multi-unit auctions with unit demand, the OEL mechanisms are the only individually undominated redistribution mechanisms that are anonymous and have linear payment functions.

Proposition 42. *For multi-unit auctions with unit demand, the OEL mechanisms are the only individually undominated redistribution mechanisms that are anonymous and have linear payment functions.*

Before proving this theorem, let us introduce the following lemma:

Lemma 4. *Let I be the set of points $(s_1, s_2, \ldots, s_k)$ ($U \geq s_1 \geq s_2 \geq \ldots \geq s_k \geq L$) that satisfy $Q_0 + Q_1 s_1 + Q_2 s_2 + \ldots + Q_k s_k = 0$ (the Q_i are constants). If the measure of I is positive (Lebesgue measure on R^k), then $Q_i = 0$ for all i.*

Proof. If $Q_i \neq 0$ for some i, then for any $U \geq s_1 \geq s_2 \geq \ldots \geq s_{i-1} \geq s_{i+1} \geq \ldots \geq s_k \geq L$, to make $Q_0 + Q_1 s_1 + Q_2 s_2 + \ldots + Q_k s_k = 0$, s_i can take at most one value. As a result the measure of I must be 0. □

Proof of Proposition 42. We still use $[\theta]_i$ to denote the ith highest type among the agents. We still use U and L to denote the upper bound and the lower bound on the agents' types, respectively.

Let r be a feasible anonymous linear redistribution mechanism. We recall that a redistribution mechanism is anonymous and linear if the redistribution function is defined as $r(\theta_{-i}) = a_0 + \sum_{j=1}^{n-1} a_j [\theta_{-i}]_j$. Here, the a_j are constants.

For any type profile, the total redistribution $\sum_{i=1}^{n} r(\theta_i) = \sum_{i=1}^{n} (a_0 + \sum_{j=1}^{n-1} a_j [\theta_{-i}]_j)$. In our setting, the total VCG payment equals $m[\theta]_{m+1}$. Therefore, for any type profile, the waste is a linear function in terms of the types. For simplicity, we rewrite the waste as $C_0 + C_1[\theta]_1 + C_2[\theta]_2 + \ldots + C_n[\theta]_n$. The C_i are constants determined by the a_i. We have

$$C_0 = -na_0$$

$$C_1 = -(n-1)a_1$$

$$C_2 = -a_1 - (n-2)a_2$$

$$C_3 = -2a_2 - (n-3)a_3$$

$$\vdots$$

$$C_m = -(m-1)a_{m-1} - (n-m)a_m$$

$$C_{m+1} = -ma_m - (n-m-1)a_{m+1} + m$$

$$C_{m+2} = -(m+1)a_{m+1} - (n-m-2)a_{m+2}$$

$$\vdots$$

$$C_{n-1} = -(n-2)a_{n-2} - a_{n-1}$$

$$C_n = -(n-1)a_{n-1}$$

Given any θ_{-i}, for any possible value of θ_i, we must have $\sum_{j=1}^{n} r(\theta_{-j}) \leq m[\theta]_{m+1}$ (non-deficit). That is, for any θ_{-i}, we have $\min_{\theta_i}(m[\theta]_{m+1} - \sum_{j=1}^{n} r(\theta_{-j})) \geq 0$. If for some θ_{-i}, we have $\min_{\theta_i}(m[\theta]_{m+1} - \sum_{j=1}^{n} r(\theta_{-j})) > \epsilon$ ($\epsilon > 0$), then we can increase $r(\theta_{-i})$ (the redistribution of agent i) by ϵ without violating the non-deficit constraint when the other agents' types are θ_{-i}. Therefore, if r is individually undominated, for any θ_{-i}, we have $\min_{\theta_i}(m[\theta]_{m+1} - \sum_{j=1}^{n} r(\theta_{-j})) = 0$.

We denote $[\theta_{-i}]_j$ (the jth highest type among θ_{-i}) by s_j ($j = 1, \ldots, n-1$). That is, $s_1 \geq s_2 \geq \ldots \geq s_{n-1}$.

$\min_{\theta_i}(m[\theta]_{m+1} - \sum_{j=1}^{n} r(\theta_{-j}))$ then equals the minimum of the following expressions:

$$\min_{L \leq \theta_i \leq s_{n-1}}(m[\theta]_{m+1} - \sum_{j=1}^{n} r(\theta_{-j}))$$

$$\min_{s_{n-1} \leq \theta_i \leq s_{n-2}}(m[\theta]_{m+1} - \sum_{j=1}^{n} r(\theta_{-j}))$$

$$\vdots$$

$$\min_{s_2 \leq \theta_i \leq s_1}(m[\theta]_{m+1} - \sum_{j=1}^{n} r(\theta_{-j}))$$

$$\min_{s_1 \leq \theta_i \leq U}(m[\theta]_{m+1} - \sum_{j=1}^{n} r(\theta_{-j}))$$

We take a closer look at $\min_{L \leq \theta_i \leq s_{n-1}}(m\theta_{m+1} - \sum_{j=1}^{n} r(\theta_{-j}))$. When $L \leq \theta_i \leq s_{n-1}$, the

jth highest type $[\theta]_j = s_j$ for $j = 1, \ldots, n-1$, and the nth highest type $[\theta]_n = \theta_i$ (this case corresponds to agent i being the agent with the lowest type). We have $\min_{L \leq \theta_i \leq s_{n-1}} (m\theta_{m+1} - \sum_{j=1}^{n} r(\theta_{-j})) = \min_{L \leq \theta_i \leq s_{n-1}} (C_0 + C_1 s_1 + C_2 s_2 + \ldots + C_{n-1} s_{n-1} + C_n \theta_i) = \min\{C_0 + C_1 s_1 + \ldots + C_{n-1} s_{n-1} + C_n L, C_0 + C_1 s_1 + \ldots + C_{n-1} s_{n-1} + C_n s_{n-1}\}$. That is, because the expression is linear, the minimum is reached when θ_i is set to either the lower bound L or the upper bound s_{n-1}.

Similarly, we have $\min_{s_{n-1} \leq \theta_i \leq s_{n-2}} (m\theta_{m+1} - \sum_{j=1}^{n} r(\theta_{-j})) = \min\{C_0 + C_1 s_1 + C_2 s_2 + \ldots + C_{n-2} s_{n-2} + C_{n-1} s_{n-1} + C_n s_{n-1}, C_0 + C_1 s_1 + C_2 s_2 + \ldots + C_{n-2} s_{n-2} + C_{n-1} s_{n-2} + C_n s_{n-1}\}$.

$\vdots$

$\min_{s_2 \leq \theta_i \leq s_1} (m\theta_{m+1} - \sum_{j=1}^{n} r(\theta_{-j})) = \min\{C_0 + C_1 s_1 + C_2 s_1 + C_3 s_2 + \ldots + C_n s_{n-1}, C_0 + C_1 s_1 + C_2 s_2 + C_3 s_2 + \ldots + C_n s_{n-1}\}$.

$\min_{s_1 \leq \theta_i \leq U} (m\theta_{m+1} - \sum_{j=1}^{n} r(\theta_{-j})) = \min\{C_0 + C_1 U + C_2 s_1 + \ldots + C_n s_{n-1}, C_0 + C_1 s_1 + C_2 s_1 + \ldots + C_n s_{n-1}\}$.

Putting all the above together, we have that for any $U \geq s_1 \geq s_2 \geq \ldots \geq s_{n-1} \geq L$, the minimum of the following expressions is 0.

- (n): $C_0 + C_1 s_1 + C_2 s_2 + \ldots + C_{n-1} s_{n-1} + C_n L$
- $(n-1)$: $C_0 + C_1 s_1 + C_2 s_2 + \ldots + C_{n-1} s_{n-1} + C_n s_{n-1}$
- $(n-2)$: $C_0 + C_1 s_1 + C_2 s_2 + \ldots + C_{n-2} s_{n-2} + C_{n-1} s_{n-2} + C_n s_{n-1}$
- $\vdots$
- (2): $C_0 + C_1 s_1 + C_2 s_2 + C_3 s_2 + \ldots + C_n s_{n-1}$
- (1): $C_0 + C_1 s_1 + C_2 s_1 + C_3 s_2 + \ldots + C_n s_{n-1}$

- (0): $C_0 + C_1 U + C_2 s_1 + C_3 s_2 + \ldots + C_n s_{n-1}$

The above expressions are numbered from 0 to n. Let $I(i)$ be the set of points $(s_1, s_2, \ldots, s_{n-1})$ ($U \geq s_1 \geq s_2 \geq \ldots \geq s_{n-1} \geq L$) that make expression (i) equal to 0. There must exist at least one i such that the measure of $I(i)$ is positive. According to Lemma 4, expression (i) must be constant at 0.

If expression (0) is constant at 0, then the waste under mechanism r is 0 whenever the highest type is equal to the upper bound U. That is, for any type profile, the waste $C_0 + C_1[\theta]_1 + C_2[\theta]_2 + \ldots + C_n[\theta]_n$ must be a constant multiple of $U - [\theta]_1$ (the waste is a linear function). We have $C_0 = -UC_1$ and $C_j = 0$ for $j \geq 2$. It turns out that the above equalities of the C_j completely determine the values of the a_j, and the corresponding mechanism is the OEL mechanism with index $k = 0$. We show this below.

We recall that the C_j satisfy the following equalities:

$$C_0 = -na_0$$

$$C_1 = -(n-1)a_1$$

$$C_2 = -a_1 - (n-2)a_2$$

$$C_3 = -2a_2 - (n-3)a_3$$

$$\vdots$$

$$C_m = -(m-1)a_{m-1} - (n-m)a_m$$

$$C_{m+1} = -ma_m - (n-m-1)a_{m+1} + m$$

$$C_{m+2} = -(m+1)a_{m+1} - (n-m-2)a_{m+2}$$

$$\vdots$$

$$C_{n-1} = -(n-2)a_{n-2} - a_{n-1}$$

$$C_n = -(n-1)a_{n-1}$$

We can solve for the value of a_{n-1} based on the value of C_n.

We can solve for the value of a_{n-2} based on the values of a_{n-1} and C_{n-1}.

$\vdots$

We can solve for the value of a_1 based on the values of a_2 and C_2.

Finally, since $C_0 = -UC_1$, the value of a_0 can also be solved.

Therefore, there is only one anonymous linear redistribution mechanism whose waste equals a constant multiple of $U - [\theta]_1$. Since the OEL mechanism with index 0 is anonymous, linear and its total waste is a constant multiple of $U - [\theta]_1$, we have that if expression (0) is constant at 0, then the mechanism must be the OEL mechanism with index 0.

Similarly, if expression (n) is constant at 0, then the waste under mechanism r is 0 whenever the lowest type is equal to the lower bound L (corresponding to the OEL mechanism with index $k = n$). If expression (i) is constant at 0 for other i, then the waste under mechanism r is 0 whenever the ith and $(i + 1)$th type equal (corresponding to the OEL mechanism with index $k = i$). This finishes the proof. □

Since being collectively undominated is strictly stronger than being individually undominated, Proposition 21 and 42 imply that for multi-unit auctions with unit demand, the family of collectively undominated VCG redistribution mechanisms that are anonymous and have linear payment functions, is exactly the family of OEL mechanisms.

Theorem 8. *For multi-unit auctions with unit demand, the family of collectively undominated VCG redistribution mechanisms that are anonymous and have linear payment functions, is exactly the family of OEL mechanisms.*

The above theorem also implies that, if we are focusing on anonymous and linear VCG redistribution mechanisms for multi-unit auctions with unit demand, then being individually undominated and being collectively undominated are equivalent.

Table 2.6: Increase in redistribution payments relative to WCO, and total VCG payments that are not redistributed, for different priority orders. Note that increases are ordered according to the priority order. The "average" item gives the average increase to the agent ordered in the kth place (first), as well as the average increase to agent i (second).

Bids	Increase	Remaining
5,13,10,8	6/7,9/7,4/7,0	2/7
5,13,8,10	6/7,9/7,0,1/7	5/7
5,10,13,8	6/7,9/7,4/7,0	2/7
5,10,8,13	6/7,9/7,0,1/7	5/7
5,8,10,13	6/7,15/7,0,0	0
5,8,13,10	6/7,15/7,0,0	0
13,5,10,8	9/7,6/7,6/7,0	0
13,5,8,10	9/7,6/7,0,0	6/7
13,10,5,8	9/7,6/7,6/7,0	0
13,10,8,5	9/7,6/7,0,6/7	0
13,8,10,5	9/7,0,0,6/7	6/7
13,8,5,10	9/7,0,6/7,0	6/7
10,13,5,8	9/7,6/7,6/7,0	0
10,13,8,5	9/7,6/7,0,6/7	0
10,5,13,8	9/7,6/7,6/7,0	0
10,5,8,13	9/7,6/7,0,0	6/7
10,8,5,13	9/7,0,6/7,0	6/7
10,8,13,5	9/7,0,0,3/7	9/7
8,13,10,5	15/7,6/7,0,0	0
8,13,5,10	15/7,6/7,0,0	0
8,10,13,5	15/7,6/7,0,0	0
8,10,5,13	15/7,6/7,0,0	0
8,5,10,13	15/7,6/7,0,0	0
8,5,13,10	15/7,6/7,0,0	0
Average (1)	1.39,0.89,0.26,0.14	0.315
Average (2)	0.71,0.64,0.64,0.70	

Table 2.7: Improving WCO using dominance techniques.

n	m	Nonanon. 3 updates	Anonymous 1 iteration	Anonymous 2 iterations
4	1	42%	66%	52%
5	1	49%	69%	55%
6	1	32%	55%	39%
5	2	44%	68%	54%
6	3	45%	68%	54%

Table 2.8: Improving Cavallo using dominance techniques.

n	Nonanon. 2 updates unknown	Anonymous 1 iteration unknown	Nonanon. 2 updates known	Anonymous 1 iteration known
5	81%	84%	61%	75%
6	76%	82%	64%	69%
7	73%	81%	54%	68%
8	78%	83%	59%	66%

Table 2.9: Improving VCG using dominance techniques.

n	Nonanon. 3 updates, unknown	Nonanon. 3 updates, known
5	88%	68%
6	91%	67%
7	95%	51%
8	96%	81%

2.4 Better Redistribution with Inefficient Allocation

So far, we have only discussed the problem of designing VCG redistribution mechanisms that are optimal in various senses. By definition, the VCG redistribution mechanisms first allocate the items efficiently and charge the VCG payments. Then, a large fraction of the VCG revenue is redistributed back to the agents, in a way that maintains the desirable properties of the original VCG mechanism, including strategy-proofness, the non-deficit property, and (sometimes) individual rationality. However, in some cases, even the best redistribution mechanism fails to redistribute a substantial amount of the VCG revenue. That is, even though the VCG redistribution mechanisms maximize *efficiency*, the total *welfare* (the sum of the agents' utilities, taking payments into account) can be very low (in fact, zero), as a result of poor redistribution. Still, the VCG redistribution mechanisms proposed earlier in this dissertation are optimal in various senses—but only under the constraint that allocation is efficient.

In this section, we study the problem of designing the allocation rule together with the redistribution scheme, allowing for the allocation to be inefficient. It turns out that even though inefficient allocation reduces efficiency, it sometimes allows for greater redistributions, so that the net effect is an increase in the sum of the agents' utilities. Moulin [84] already provided an example where inefficient allocation can lead to better results, but left a more thorough investigation for future research. As we will see, the example mechanism that he proposed will turn out to be useful for us.

In Subsection 2.4.1, we cover some basic definitions. In Subsection 2.4.2, we define a class of allocation mechanisms that we call *linear allocation mechanisms*, and propose an optimization model for simultaneously finding an allocation mechanism and a payment redistribution rule which together are optimal, given that the allocation mechanism is required to be either one of, or a mixture over, a finite set of specified linear allocation mechanisms. In Subsection 2.4.3 to Subsection 2.4.4, we propose several specific mechanisms that are based on burning items, excluding agents, and (most generally) partitioning the items and agents into groups. We show or conjecture that these mechanisms are optimal among various classes of mechanisms.

2.4.1 Formalization

We will restrict our attention to multi-unit auctions with unit demand. We still use n and m to denote the number of agents and the number of units. Since we are dealing with the unit demand setting, we assume $n > m$. (Otherwise, it is clearly optimal to give every agent a unit and charge nothing.) As usual, for the ith agent, we denote her true/reported type/bid for winning one unit by v_i (we are restricting our attention to strategy-proof mechanisms). Without losing generality, we assume that $v_1 \geq v_2 \geq \ldots \geq v_n \geq 0$. A *bid profile* is a vector $V = (v_1, v_2, \ldots, v_n)$.

Let M be a strategy-proof allocation mechanism. (We use "allocation mechanism" to refer to the mechanism before redistribution—for example, in a VCG redistribution mechanism, the VCG mechanism is the allocation mechanism, whereas the complete mechanism also includes the redistributions.) M does not need to be deterministic: in general, M can be a probability mixture over t deterministic mechanisms $M_1, \ldots, M_t$. (When $t = 1$, M is deterministic.) With probability p_i, mechanism M_i is chosen ($\sum_{i=1}^{t} p_i = 1$).

For each bid profile V, we define $U_M(V)$ to be the total efficiency (sum of obtained valuations) that results under M for V (this does not take payments into account). We have $U_M(V) = \sum_{i=1}^{t} p_i U_{M_i}(V)$. Similarly, let $P_M(V)$ be the total revenue (sum of the agents' payments) that results under M for V. We have $P_M(V) = \sum_{i=1}^{t} p_i P_{M_i}(V)$.

For multi-unit auctions with unit demand, the VCG mechanism is just the $(m+1)th$ price auction: the agents with the m highest bids each win one unit and each pay the value of the $(m+1)$th-highest bid. Hence, if M is the VCG mechanism, $U_M(V) = \sum_{i=1}^{m} v_i$ and $P_M(V) = mv_{m+1}$.

Given a strategy-proof allocation mechanism M and a bid profile V, without redistribution, the agents' welfare under M equals $U_M(V) - P_M(V)$. The welfare can potentially be increased by introducing redistribution payments. We require that the redistribution payment to each agent is independent of her own bid, so that the mechanism will remain strategy-proof. That is, agent i receives a redistribution payment $R(V_{\sim i})$, where $V_{\sim i}$ is the bid profile without v_i $(v_1, \ldots, v_{i-1}, v_{i+1}, \ldots, v_n)$, and R is any real-valued function.

Not all redistribution functions (R) are feasible. For a redistribution mechanism to be feasible, we require two additional properties. First, we require that $P_M(V) \geq \sum_{i=1}^{n} R(V_{\sim i})$ for all V. That is, the mechanism must satisfy the *non-deficit* property: the total redistribution should never exceed the revenue collected by M. Otherwise, we need external funds to subsidize the mechanism.

We also require that the mechanism be *(ex-post) individually rational*: if M is deterministic, then for any bid profile V, every agent's utility after redistribution must be nonnegative. If M is not deterministic, then for any bid profile V, every agent's expected utility after redistribution must be nonnegative.

With redistribution, for a bid profile V, the agents' welfare is $U_M(V) - P_M(V) + \sum_{i=1}^{n} R(V_{\sim i})$. Our goal is to find a strategy-proof allocation mechanism M and a redistribution function R that are feasible and maximize this expression. However, this is not a well-defined objective, because the value of this expression depends on V. It could be that one choice of M and R maximizes the expression for some V, while another choice of M and R maximizes the expression for another V. In this section, we return to the worst-case perspective, as in Section 2.1. Specifically, consider an omnipotent *perfect* allocation mechanism that magically identifies the agents with the m highest true valuations, without asking for their bids, and allocates the units to these agents at no charge. Clearly this mechanism obtains the largest welfare that we could hope for (without deficits). Our objective is to design mechanisms that are *competitive* with this perfect allocation mechanism. We say a redistribution mechanism (M, R) is α-*competitive* against the perfect mechanism if the agents' welfare under (M, R) is at least $\alpha \sum_{i=1}^{m} v_i$, for all bid profiles V. ($\sum_{i=1}^{m} v_i$ is the agents' welfare under the perfect mechanism.) Our objective is to find the redistribution mechanism (M, R) that is the most competitive, that is, that maximizes α, while satisfying the individual rationality and non-deficit properties.

In what follows, we construct a motivational example feasible strategy-proof mechanism that allocates inefficiently and has a higher competitive ratio (with the perfect mechanism) than all feasible strategy-proof mechanisms that always maximize efficiency. Actually, it was shown in [84] that the WCO mechanism has the highest competitive ratio among all feasible strategy-proof mechanisms that always maximize efficiency.

Proposition 43. *[84] The WCO mechanism has the highest competitive ratio α against the perfect allocation mechanism, among all (efficient) VCG redistribution mechanisms that are individually rational and satisfy the non-deficit property.*

When $n = 3$ and $m = 2$, the WCO mechanism is not competitive at all: $\alpha_{WCO}(3, 2) = 0$. In contrast, the following simple mechanism that allocates inefficiently is somewhat competitive:

- Burn (throw away) one unit.
- Allocate the remaining unit according to the WCO mechanism for $n = 3$ and $m = 1$.

The new mechanism is feasible and strategy-proof because it is equivalent to the WCO mechanism for $n = 3$ and $m = 1$. It is not efficient because one unit is burned. Since $\alpha_{WCO}(3, 1) = \frac{1}{3}$, the new mechanism is $\frac{1}{3}$-competitive against the perfect allocation mechanism for one unit ($m = 1$). That is, the new mechanism guarantees a welfare of $\frac{1}{3}v_1$ for any bid profile V. Since $v_1 \geq v_2$, it also guarantees a total utility of $\frac{1}{6}(v_1 + v_2)$ for all bid profiles. Hence, the competitive ratio of the new mechanism against the perfect allocation mechanism for two units ($m = 2$) is at least $\frac{1}{6}$. That is, this new mechanism has a higher competitive ratio than any VCG redistribution mechanism (any feasible strategy-proof mechanism that allocates the units efficiently). So, ironically, in some cases, the agents are happier if one unit is burned. Motivated by this example, in what follows, we study mechanisms that allocate inefficiently (and in Subsection 2.4.3, we specifically study mechanisms that are based on burning units).

2.4.2 Linear allocation mechanisms

In this subsection, we define a class of mechanisms that we call *linear allocation mechanisms*. We then provide a general technique for finding the optimal redistribution function for any given linear allocation mechanism. We also show how to simultaneously find the optimal linear allocation mechanism and the corresponding redistribution function, given that the allocation mechanism is required to be one of, or a mixture over, a finite set of specified linear allocation mechanisms.

Definition 3. A (strategy-proof) allocation mechanism M is *linear* if the following two conditions are satisfied:

- (linearity) $U_M(V)$ and $P_M(V)$ are linear combinations of the v_i.
- (normalized individual rationality) M is individually rational, and an agent's payment is always 0 if her bid is 0.

Example 13. The VCG mechanism is linear, for the following reasons. In the VCG mechanism, the agents with the highest m bids each win one unit and each pay the value of the $(m+1)$th-highest bid. That is, for any bid profile $V = (v_1, v_2, \ldots, v_n)$, $U_M(V) = \sum_{i=1}^{m} v_i$ and $P_M(V) = mv_{m+1}$, which are both linear.[17] The normalized individual rationality condition is also satisfied by the VCG mechanism. Under the VCG mechanism, the payment from an agent is always less than or equal to her own bid, and is never negative. When an agent's bid is 0, her payment must be 0.

Example 14. The *random allocation mechanism* in which the winners are picked uniformly at random (without replacement), and there are no payments, is linear, for the following reasons. Under this mechanism, for any bid profile V, $U_M(V) = \frac{m}{n}\sum_{i=1}^{n} v_i$, and $P_M(V) = 0$, which are both linear. The normalized individual rationality condition is also satisfied.

Proposition 44. *Any probability mixture over linear allocation mechanisms is also linear.*

Proof. Let M be a mixture over t linear allocation mechanisms $M_1, M_2, \ldots, M_t$, where M_i is chosen with probability p_i. We have $U_M(V) = \sum_{i=1}^{t} p_i U_{M_i}(V)$ and $P_M(V) = \sum_{i=1}^{t} p_i P_{M_i}(V)$, which are both linear, because for any i, U_{M_i} and P_{M_i} are linear. Normalized individual rationality also holds: because $M_1, M_2, \ldots, M_t$ are all individually rational, any mixture over them is also individually rational. If an agent's bid is 0, then for all i, her payment under M_i is 0. This implies that her payment under any mixture over the M_i is also 0. □

Given the number of agents n, the number of units m, and a linear allocation mechanism M, the following optimization model can be used to find an optimal

[17] We emphasize that the linearity depends on the fact that the bids are sorted. In fact, if we increase the $(m+1)$th-highest bid, then the revenue will increase, but only up to the point where the bid equals the mth-highest bid; if we increase the bid further, the revenue will not change. So in this sense, the VCG mechanism is not linear in the bids, but this is not the type of linearity that is used in the definition.

redistribution function R, so that the resulting mechanism (M, R) has the highest competitive ratio. That is, we are computing the optimal redistribution function for a fixed allocation mechanism.

Variable function: $R : [0, \infty)^{n-1} \to \mathbb{R}$
Variable: α
Maximize α
Subject to:
For every bid profile $V = (v_1, v_2, \ldots, v_n)$ with $v_1 \geq v_2 \geq \ldots \geq v_n \geq 0$
$R(V_{\sim n}) \geq 0$ (individual rationality)
$P_M(V) \geq \sum_{i=1}^{n} R(V_{\sim i})$ (non-deficit)
$U_M(V) - P_M(V) + \sum_{i=1}^{n} R(V_{\sim i}) \geq \alpha \sum_{i=1}^{m} v_i$ (competitive ratio constraint)

Because M is linear, it satisfies the normalized individual rationality condition. Hence, if the agent with the lowest bid bids 0, her payment under M must be 0. Such an agent's utility is 0 when there is no redistribution, regardless of whether she wins a unit or not. With redistribution, such an agent's utility is just her redistribution $R(V_{\sim n})$. Therefore, for the resulting mechanism (M, R) to satisfy the individual rationality constraint, it is necessary that $R(V_{\sim n}) \geq 0$ for all $V_{\sim n}$. Since $R(V_{\sim n})$ does not depend on the value of v_n, equivalently, it is necessary that $R(V_{\sim n}) \geq 0$ for all V. Conversely, if $R(V_{\sim n}) \geq 0$ for all V, then the function R is always nonnegative, because for any $x_1 \geq x_2 \geq \ldots \geq x_{n-1}$ there exists a V such that $V_{\sim n} = (x_1, x_2, \ldots, x_{n-1})$. This implies that $R(V_{\sim n}) \geq 0$ for all V is also a sufficient condition for individual rationality, because M is individually rational without redistribution and nonnegative redistribution never decreases an agent's utility. This is why the individual rationality constraint can be written as $R(V_{\sim n}) \geq 0$ for all V.

Now, suppose that the allocation mechanism is not fixed; specifically, suppose that we need to choose one mechanism M from a set of t linear allocation mechanisms $\{M_1, \ldots, M_t\}$, so that M, coupled with a corresponding optimal redistribution function, has the highest competitive ratio. Then the optimization model becomes:

Variable function: $R : [0, \infty)^{n-1} \rightarrow \mathbb{R}$
Variable: α
Binary variables: $p_1, p_2, \ldots, p_t$
Maximize α
Subject to:
For every bid profile $V = (v_1, v_2, \ldots, v_n)$ with $v_1 \geq v_2 \geq \ldots \geq v_n \geq 0$
$R(V_{\sim n}) \geq 0$ (individual rationality)
$\sum_{j=1}^{t} p_j P_{M_j}(V) \geq \sum_{i=1}^{n} R(V_{\sim i})$ (non-deficit)
$\sum_{j=1}^{t} p_j U_{M_j}(V) - \sum_{j=1}^{t} p_j P_{M_j}(V) + \sum_{i=1}^{n} R(V_{\sim i}) \geq \alpha \sum_{i=1}^{m} v_i$ (competitive ratio constraint)
$\sum_{j=1}^{t} p_j = 1$

It might not be clear, at first glance, why we would want to introduce the binary variables p_j, rather than just solve the original model t times. The reason is that if we change the p_j into continuous variables ranging from 0 to 1, then the modified model optimizes for the best allocation mechanism among mechanisms that are mixtures over $\{M_1, M_2, \ldots, M_t\}$ (and simultaneously, it optimizes the corresponding redistribution function). We call the above optimization model in which the p_j are binary the discrete model (DM), and we call the modified optimization model the continuous model (CM).

DM and CM both optimize over functions, not just variables. However, as it turns out, an optimal solution can be found by means of a linear program (for CM) or a mixed integer program (for DM). The linear/mixed integer program can be solved directly, using any solver.

The constraints of DM and CM must be satisfied for any bid profile $V = (v_1, v_2, \ldots, v_n)$ with $v_1 \geq v_2 \geq \ldots \geq v_n \geq 0$. We use $z \in \{0, \ldots, n\}$ to denote the bid profile in which the highest z bids are 1 and the remaining bids are 0. If we only require that the constraints be satisfied for bid profiles from 0 to n, then the objective value should be greater than or equal to the original objective value. The relaxed optimization models (UCM and UDM, for "upper bounding continuous/discrete model") are:

Variable function: $R : [0, \infty)^{n-1} \to \mathbb{R}$
Variables: $p_1, p_2, \dots, p_t \geq 0$ **(UCM only)**, α
Binary variables: $p_1, p_2, \dots, p_t$ **(UDM only)**
Maximize α
Subject to:
For every bid profile V from 0 to n
$R(V_{\sim n}) \geq 0$ (individual rationality)
$\sum_{j=1}^{t} p_j P_{M_j}(V) \geq \sum_{i=1}^{n} R(V_{\sim i})$ (non-deficit)
$\sum_{j=1}^{t} p_j U_{M_j}(V) - \sum_{j=1}^{t} p_j P_{M_j}(V) + \sum_{i=1}^{n} R(V_{\sim i}) \geq \alpha \sum_{i=1}^{m} v_i$ (competitive ratio constraint)
$\sum_{j=1}^{t} p_j = 1$

Effectively, UCM is a linear program and UDM is a mixed integer program. If $V = z$, then $V_{\sim i}$ (the bids other than i's own bid) contains z copies of 1 and $n-1-z$ copies of 0 for $i > z$, and $V_{\sim i}$ contains $z-1$ copies of 1 and $n-z$ copies of 0 for $i \leq z$. Let us denote $R(V_{\sim i})$ by R_x if $V_{\sim i}$ contains x copies of 1 ($0 \leq x \leq n-1$). Then, the n variables $R_0, R_1, \dots, R_{n-1}$ specify everything about the redistribution function that affects the UDM/UCM programs; thus, they are the only variables that we need, in addition to the p_j and α. The $P_{M_j}(V)$ and $U_{M_j}(V)$ are constants that we need to evaluate for V from 0 to n. The constraints and the objective function are all linear. This results in the following linear/mixed integer program:

Variables: $p_1, p_2, \dots, p_t \geq 0$ **(UCM only)**
α, $R_0, R_1, \dots, R_{n-1}$
Binary variables: $p_1, p_2, \dots, p_t$ **(UDM only)**
Maximize α
Subject to:
$\sum_{j=1}^{t} p_j = 1$
$R_0 = 0$, $R_x \geq 0$ for $1 \leq x \leq n-1$
$\sum_{j=1}^{t} p_j P_{M_j}(x) \geq x R_{x-1} + (n-x) R_x$ for $1 \leq x \leq n$
$\sum_{j=1}^{t} p_j U_{M_j}(x) - \sum_{j=1}^{t} p_j P_{M_j}(x) + x R_{x-1} + (n-x) R_x \geq \alpha \min\{x, m\}$ for $1 \leq x \leq n$

Let α_C^*, $R_{C,0}^*, R_{C,1}^*, \dots, R_{C,n-1}^*$, and $p_{C,1}^*, p_{C,2}^*, \dots, p_{C,t}^*$ denote an optimal solution to UCM; similarly, let α_D^*, $R_{D,0}^*, R_{D,1}^*, \dots, R_{D,n-1}^*$, and $p_{D,1}^*, p_{D,2}^*, \dots, p_{D,t}^*$ denote an optimal solution to UDM. We know that α_C^* (α_D^*) is an upper bound on the competitive ratio that can be obtained in the continuous (discrete) case; we will show that, in fact, α_C^* (α_D^*) can be obtained, so that it is the optimal competitive ratio. The following theorem shows how to convert the optimal solution to UCM (UDM) into a redistribution mechanism that is defined for all V and that obtains competitive ratio α_C^* (α_D^*).

Theorem 9. *The optimal objective value for CM (DM) equals α_C^* (α_D^*). For DM, an optimal allocation mechanism is M_j, where j is the (only) index that satisfies $p_{D,j}^* = 1$. For CM, an optimal allocation mechanism is the mixture over $M_1, M_2, \ldots, M_t$ where M_j is chosen with probability $p_{C,j}^*$.*

An optimal redistribution function R_C can be obtained from the $R_{C,x}^*$ as follows: for any V and any i,

$$R_C(V_{\sim i}) = R_{C,0}^* V_{\sim i}(1) + \sum_{x=1}^{n-1}(R_{C,x}^* - R_{C,x-1}^*)V_{\sim i}(x)$$

Here, $V_{\sim i}(x)$ is the xth-highest bid among bids other than i's own bid. An optimal redistribution function R_D is defined similarly.

We note that when $V_{\sim i}$ consists of z ones and $n-z-1$ zeroes, we have $R_C(V_{\sim i}) = R_{C,0}^* + \sum_{x=1}^{z}(R_{C,x}^* - R_{C,x-1}^*) = R_{C,z}^*$ (if $z = 0$, $R_C(V_{\sim i}) = 0 = R_{C,0}^*$). (In a sense, R is an interpolation of these values.) Before proving the theorem, we give the following lemma. We presented a similar lemma in Section 2.1, namely, Lemma 1.

Lemma 5. *When the c_i do not depend on the x_i, the following two systems of inequalities are equivalent:*

(a) $c_1x_1 + c_2x_2 + \ldots + c_sx_s \geq 0$ *for all* $x_1 \geq x_2 \geq \ldots \geq x_s \geq 0$.

(b) $c_1x_1 + c_2x_2 + \ldots + c_sx_s \geq 0$ *for all* $x_1 \geq x_2 \geq \ldots \geq x_s$*, where each* $x_i \in \{0, 1\}$.

Proof. (a) $\Rightarrow$ (b) is trivial. We now prove (b) $\Rightarrow$ (a). (b) implies that $\sum_{i=1}^{j} c_i \geq 0$ for all $1 \leq j \leq s$. We have $c_1x_1 + c_2x_2 + \ldots + c_sx_s = \sum_{j=1}^{s-1}(\sum_{i=1}^{j} c_i)(x_j - x_{j+1}) +$

$(\sum_{i=1}^{s} c_i)x_s$. For all $x_1 \geq x_2 \geq \ldots \geq x_s \geq 0$, each term of the above expression is nonnegative, hence the whole expression is nonnegative. So (b) $\Rightarrow$ (a). □

Now we are ready to prove Theorem 9.

Proof. We only need to prove that the solution described in the theorem is a feasible solution for CM (DM). (We emphasize that feasibility also entails obtaining the competitive ratio α_C^* (α_D^*) everywhere.) Because it is feasible, it is also optimal, because α_C^* (α_D^*) is an upper bound on CM (DM).

In the proposed solution, we have $R_C(V_{\sim i}) = R_{C,0}^* V_{\sim i}(1) + \sum_{x=1}^{n-1}(R_{C,x}^* - R_{C,x-1}^*)V_{\sim i}(x)$. For specific i and x, when $x < i$, $V_{\sim i}(x) = v_x$, and when $x \geq i$, $V_{\sim i}(x) = v_{x+1}$. (We recall that $V_{\sim i}(x)$ is the xth-highest bid among bids other than i's own bid.) Hence, $R_C(V_{\sim i})$ is linear in $v_1, v_2, \ldots, v_n$, where the coefficients are determined by the constants $R_{C,x}^*$ (we have similar results for R_D). For all $1 \leq j \leq t$, U_{M_j} and P_{M_j} are both linear in $v_1, v_2, \ldots, v_n$ by the linearity assumption. So for all of the constraints in CM (DM), with the exception of the probability constraint, each side of the inequality is a linear combination of the v_i.

We need to prove that these constraints are satisfied for all $v_1 \geq v_2 \geq \ldots \geq v_n \geq 0$. By Lemma 5, we only need them be satisfied for all $v_1 \geq v_2 \geq \ldots \geq v_n \geq 0$ where the v_i are binary variables. That is, we only need them be satisfied for the bid profiles V from 0 to n. But for these V, the constraints of CM (DM) are identical with the constraints of UCM (UDM), because, as we already noted, the function R_C (R_D) that we have defined coincides with the $R_{C,z}^*$ ($R_{D,z}^*$) on these V. □

Using Theorem 9, given the number of agents n and the number of units m, we can find the optimal allocation mechanism M, and simultaneously, a corresponding optimal redistribution function R, so that the resulting mechanism (M, R) maximizes the competitive ratio—under the constraint that M must be one of, or a mixture over, a finite set of specific linear allocation mechanisms $M_1, M_2, \ldots, M_t$.

2.4.3 Burning units

In this subsection, we study allocation mechanisms that are based on (sometimes) burning units. As was showed earlier, in some cases we can achieve a higher competitive ratio by burning units than by using the most competitive mechanism that is feasible, strategy-proof and efficient (the WCO mechanism).

We start by characterizing a set of mechanisms based on the idea of burning units. First, we construct m allocation mechanisms that are based on burning a deterministic number of units. Let M_i ($i = 1 \ldots m$) be the allocation mechanism in which $m - i$ units are burned, and the remaining i units are allocated efficiently according to the VCG mechanism. M_m is just the original VCG mechanism. We note that it makes no sense to burn all units, hence $i > 0$. We call the M_i *deterministic burning allocation mechanisms.* We can also construct allocation mechanisms in which a random number of units are burned, by randomizing over the M_i. Let M be a mixture of the M_i, where mechanism M_i is chosen with probability p_i. That is, M is the mechanism in which with probability p_i, exactly $m - i$ units are burned. (If $p_i = 1$ for some i, then M is just M_i.) We call such mixtures over the M_i *randomized burning allocation mechanisms.*

The deterministic burning allocation mechanisms are strategy-proof, because the remaining units are allocated according to the VCG mechanism, which is strategy-proof. It follows that the randomized burning allocation mechanisms are also strategy-proof. Also, the deterministic burning allocation mechanisms are linear. When there are i units remaining (M_i), the agents with the i highest bids each win one unit, and each pay the value of the $(i+1)$-th highest bid. That is, for any bid profile $V = (v_1, v_2, \ldots, v_n)$, $U_{M_i}(V) = \sum_{j=1}^{i} v_j$ and $P_{M_i}(V) = iv_{i+1}$. Both U_{M_i} and P_{M_i} are linear in the v_i. The normalized individual rationality condition is also satisfied. Therefore, the deterministic burning allocation mechanisms are linear. By Proposition 44, we also have that the randomized burning allocation mechanisms are linear.

Using Theorem 9, we can find an optimal allocation mechanism M, and a corresponding optimal redistribution function R, so that (M, R) maximizes the competitive ratio, given that M is one of the deterministic burning allocation mechanisms, or M is a randomized burning allocation mechanism.

In Table 2.10, we present the results for different numbers of agents and different numbers of units. The second column (α_D^*) gives the optimal competitive ratio among all feasible mechanisms (M, R) where M is one of the deterministic burning allocation mechanisms. The integers in the third column are the number of units burned in the optimal mechanism that corresponds to α_D^*. The fourth column (α_C^*) is the optimal competitive ratio among all feasible mechanisms (M, R) where M is a randomized burning allocation mechanism. The values in the fifth column are the probabilities of having one unit burned in the optimal mechanism that corresponds to α_C^*. (It turns out that in the optimal mechanism, either exactly one unit is burned with a certain probability, or nothing is burned.) Finally, as a benchmark, the sixth column (α_{WCO}^*) gives the competitive ratio of the WCO mechanism (the optimal competitive ratio among all feasible mechanisms (M, R) where M allocates efficiently).

For the case of $n = 10$, $m = 1, \ldots, 9$, we compare the values of α_{WCO}^*, α_D^* and α_C^* in Figure 2.3. When m is small, the three values are the same. As m gets large, the value of α_{WCO}^* decreases all the way to 0; the value of α_D^* also decreases but it gets stable when its value goes down to around 0.5; the value of α_C^* first decreases, but then increases, at the end almost reaches 1.

Of course, $\alpha_{WCO}^* \leq \alpha_D^* \leq \alpha_C^*$; it turns out that all of these inequalities are sometimes strict. Therefore, in general we need to burn a random number of units to get the most competitive redistribution mechanism.

While we can use Theorem 9 in this way to solve for the most competitive redis-

Table 2.10: Competitive ratios of burning allocation mechanisms.

	α_D^*	burn	α_C^*	burn	α_{WCO}^*
n=4,m=1	0.571	0	0.571	0	0.571
n=4,m=2	0.286	1	0.667	0.67	0.250
n=4,m=3	0.267	2	0.889	0.33	0
n=6,m=1	0.839	0	0.839	0	0.839
n=6,m=3	0.410	1	0.800	0.60	0.375
n=6,m=5	0.356	3	0.960	0.20	0
n=8,m=1	0.945	0	0.945	0	0.945
n=8,m=3	0.646	0	0.762	0.71	0.646
n=8,m=5	0.452	2	0.914	0.43	0.276
n=8,m=7	0.422	4	0.980	0.14	0

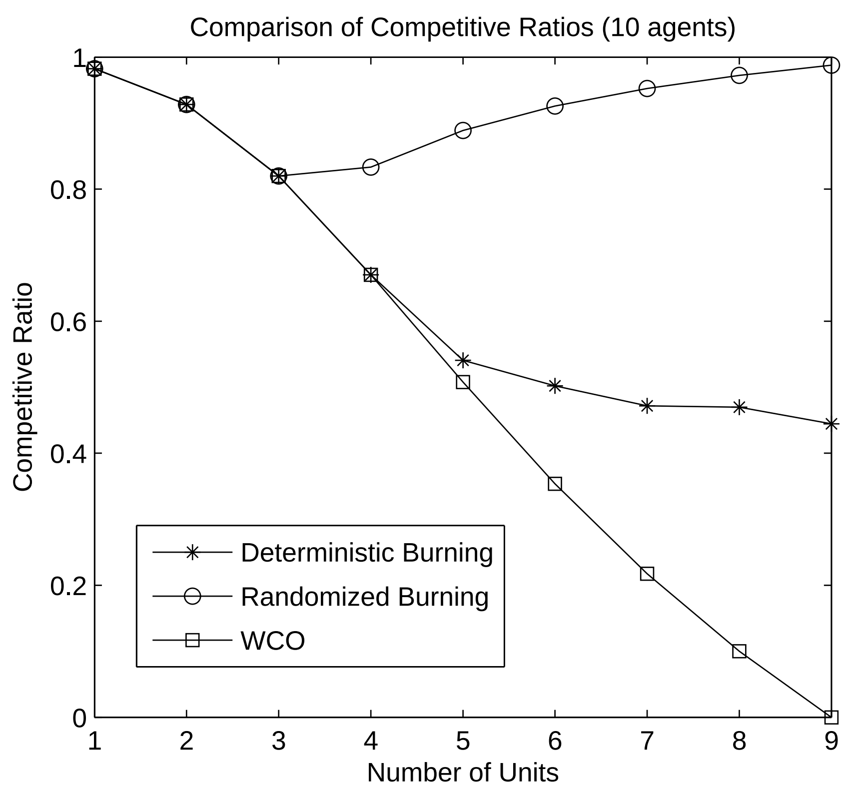

FIGURE 2.3: A comparison of competitive ratios of deterministic burning allocation mechanisms, random burning allocation mechanisms, and WCO mechanisms.

tribution mechanism in this class of mechanisms for any given n and m, it would be nice to have a general analytical characterization of the most competitive redistribution mechanism. The following proposition specifies a burning-based redistribution mechanism for each m, n pair, and gives the competitive ratio for these mechanisms. We conjecture that these mechanisms are in fact the most competitive in this class of mechanisms (including the randomized mechanisms), but have not been able to prove it. (However, using the linear programming methodology from Theorem 9, we have verified that this conjecture is true for all $n \leq 10$.)

Proposition 45. *Given n and m, using a redistribution mechanism (M, R) where M is a randomized burning allocation mechanism, we can achieve the following competitive ratio:*

$$\max\{1 - \frac{\binom{n-1}{m}}{\sum_{j=m}^{n-1} \binom{n-1}{j}}, \frac{mn-n}{mn-m}\}$$

If the first expression is greater (or equal), then the mechanism achieving the above ratio is the worst-case optimal VCG redistribution mechanism (nothing is burned).

If the second expression is greater, then the mechanism achieving the above ratio is the following:

- *Burn (throw away) one unit with probability $\frac{n-m}{n-1}$.*
- *The remaining units are allocated according to the VCG mechanism.*
- *After the VCG payments, every agent receives a redistribution payment of $\frac{m-1}{n-1}$ times the m-th highest bid among bids other than this agent's own bid. (Unlike the VCG payments, the redistribution does not depend on whether a unit was burned.)*

Proof. We already know that the WCO mechanism is strategy-proof, feasible, and has competitive ratio $1 - \frac{\binom{n-1}{m}}{\sum_{j=m}^{n-1} \binom{n-1}{j}}$. Hence, we only need to show that the other mechanism proposed in the proposition is strategy-proof, feasible, and has competitive ratio $\frac{mn-n}{mn-m}$, for the values of m, n for which this mechanism outperforms the WCO mechanism.

If $m = 1$, then $\frac{mn-n}{mn-m} = 0$, which cannot be greater than the competitive ratio of the WCO mechanism. So we only need to consider $m > 1$. We have already proved that any randomized burning allocation mechanism is strategy-proof. After introducing redistribution, the mechanism remains strategy-proof, because the redistribution does not depend on the agent's own bid. Individual rationality is satisfied because the randomized burning allocation mechanism is individually rational, and the redistribution is always nonnegative. For any bid profile V, the VCG revenue is mv_{m+1} when nothing is burned, and the VCG revenue is $(m-1)v_m$ when one unit is burned. Together, the expected[18] VCG revenue is $\frac{n-m}{n-1}(m-1)v_m + (1 - \frac{n-m}{n-1})mv_{m+1} = \frac{m-1}{n-1}(mv_{m+1} + (n-m)v_m)$. For the agents bidding $v_1, \ldots, v_m$, the redistribution received is $\frac{m-1}{n-1}v_{m+1}$. For the other agents, the redistribution received is $\frac{m-1}{n-1}v_m$. Therefore, the total redistribution equals the total VCG payment, so the non-deficit criterion is satisfied. We conclude that the mechanism is feasible.

Now we show that the mechanism has competitive ratio $\frac{mn-n}{mn-m}$. With probability $\frac{n-m}{n-1}$, the total efficiency is $\sum_{i=1}^{m-1} v_i$ (one unit is burned). When nothing is burned, the total efficiency is $\sum_{i=1}^{m} v_i$. In expectation, the total efficiency is $\frac{n-m}{n-1}\sum_{i=1}^{m-1} v_i + (1 - \frac{n-m}{n-1})\sum_{i=1}^{m} v_i$. This is greater than or equal to $\frac{n-m}{n-1}\frac{m-1}{m}\sum_{i=1}^{m} v_i + (1 - \frac{n-m}{n-1})\sum_{i=1}^{m} v_i =$

[18] The mechanism satisfies the non-deficit criterion only *in expectation* over the choice of whether to burn a unit. Alternatively, we can charge each agent her *expected* VCG payment, in which case there will certainly be no deficit. One may worry that this will result in individual rationality only holding in expectation. However, interestingly, individual rationality continues to hold unconditionally if we charge the expected VCG payment: the only agent that faces any randomness is the mth agent, and she pays $(1 - \frac{n-m}{n-1})v_{m+1}$, but then receives a redistribution of $\frac{m-1}{n-1}v_{m+1}$, for a total payment of 0, so that she is not unhappy even if the unit is thrown away. However, in this case, she would prefer to place a bid of 0 instead—so the resulting mechanism is strategy-proof only in expectation over the mechanism's random choice. (In contrast, the mechanism in Proposition 45 is unconditionally strategy-proof and individually rational.)

$\frac{mn-n}{mn-m} \sum_{i=1}^{m} v_i$ (we have equality when all v_i are equal). Since the total payment equals the total redistribution, efficiency is equal to welfare, so we conclude that we obtain the competitive ratio $\frac{mn-n}{mn-m}$. □

Conjecture 1. *The competitive ratio in Proposition 45 is optimal for mechanisms (M, R) where M is a randomized burning allocation mechanism. That is, if M is required to be a randomized burning allocation mechanism, then there is an optimal mechanism that either never burns anything (so that it coincides with the WCO mechanism), or burns exactly one unit with some probability, so that all the revenue can be redistributed.*

2.4.4 Partitioning units and agents

In this subsection, we study allocation mechanisms that are based on partitioning the units and the agents into groups. This is an idea that has previously been proven effective in mechanism design [9, 50]. Based on this idea, we first characterize a class of strongly budget balanced allocation mechanisms (in the setting of multi-unit auctions with unit demand). Some of the mechanisms in this class have been proposed previously [45, 84]. We focus on finding the most competitive mechanism in this class. Because all of the mechanisms in this class are strongly budget balanced, there will be no redistributions.

We start with two example mechanisms. They are both based on excluding one individual agent from the set of all agents. The first one is due to Moulin [84], and the second one is due to Faltings [45].

Example Mechanism 1

- Exclude one agent from the auction, uniformly at random.
- Assign one unit to the excluded agent at no charge.

- The remaining units are allocated to the remaining agents according to the VCG mechanism.

- Transfer all the VCG revenue to the excluded agent.

Example Mechanism 2

- Exclude one agent from the auction, uniformly at random.

- Units are allocated to the remaining agents according to the VCG mechanism.

- Transfer all the VCG revenue to the excluded agent.

Both example mechanisms are strategy-proof, individually rational and strongly budget balanced. In the first mechanism, one agent is excluded and assigned one unit. In the second mechanism, one agent is excluded and assigned zero units.

We now introduce our class of mechanisms that is based on partitioning the agents; this class generalizes both of the previous two mechanisms.

Definition 4. Given n and m, for $n_1 \in \{1, \ldots, \lfloor \frac{n}{2} \rfloor\}$, $m_1 \in \{0, \ldots, \min\{n_1, m\}\}$, we define the following mechanism:

- Pick n_1 agents to form one group, uniformly at random. The other $n - n_1$ agents form the second group.

- Allocate m_1 units among the first group, according to the VCG mechanism.

- Allocate the remaining $m - m_1$ units among the second group, according to the VCG mechanism.

- Transfer the VCG revenue from the first group to the second group, in any predetermined way.

- Transfer the VCG revenue from the second group to the first group, in any predetermined way.

We call this mechanism the (n_1, m_1)*-partition mechanism.*

We note that Example Mechanisms 1 and 2 are the $(1, 1)$-partition mechanism and the $(1, 0)$-partition mechanism, respectively.

Proposition 46. *The partition mechanisms are strategy-proof, individually rational, and strongly budget balanced.*

Proof. Without transferring the VCG revenue, every agent is participating in a VCG mechanism, which must be strategy-proof. For each agent, the transfer payment she receives depends only on the bids from the other group of agents, hence it does not affect her incentives. Therefore, the mechanisms are strategy-proof. Similarly, without transferring the VCG revenue, every agent is participating in a VCG mechanism, which must be individually rational. With transferring, the agents' utilities become higher or stay the same. Therefore, the mechanisms are individually rational. Finally, the strong budget balance property follows from the fact that the entire VCG revenue is transferred. □

Since the partition mechanisms are strongly budget balanced, welfare must equal efficiency. Hence, for our objective of finding the most competitive partition mechanism, we can completely ignore the VCG payments and the revenue transferring process. That is, for the analysis that follows, we pretend that there are no payments of any kind; when we use the mechanism, we add the VCG payments and transfers back to achieve strategy-proofness. We now show that by ignoring the payments and transfers, the partition mechanisms become linear mechanisms (albeit linear mechanisms that are not strategy-proof, but this does not matter). Given n and m, let M_{n_1,m_1} be the (n_1, m_1)-partition mechanism. For any bid profile $V = (v_1, v_2, \ldots, v_n)$, under M_{n_1,m_1}, there is only a finite number of ways of dividing the v_i into two groups of size n_1 and $n - n_1$ (and each of these ways receives equal probability). For any specific way of dividing, the agents' total efficiency is linear in the v_i. Since each way of dividing happens with equal probability, the expected total efficiency is also linear in the v_i. That is, $U_{M_{n_1,m_1}}(V)$ is linear in the v_i. We also have $P_{M_{n_1,m_1}} = 0$. Hence, the partition mechanisms satisfy the linearity condition. The normalized individual rationality condition is also satisfied (after ignoring the VCG payments and the revenue transferring process). Thus, we can use the general technique in Theorem 9 to solve for the optimal partition mechanism. However, we now present a simpler solution technique based on the special structure of the class of partition mechanisms.

The following proposition characterizes the competitive ratio of a given partition mechanism.

Proposition 47. *Given n and m, the competitive ratio of the (n_1, m_1)-partition mechanism equals*

$$\frac{U_{M_{n_1,m_1}}(m)}{m}$$

Here, $U_{M_{n_1,m_1}}(m)$ is the expected efficiency (welfare) under the (n_1, m_1)-partition mechanism when m agents bid 1 and the remaining agents bid 0. This competitive ratio is equal to

$$\frac{\sum_{x\in X}\binom{n_1}{x}\binom{n-n_1}{m-x}(\min\{x, m_1\} + \min\{m - x, m - m_1\})}{m\binom{n}{m}}$$

Here, $X = \{x|0 \leq x \leq n_1, 0 \leq m - x \leq n - n_1\}$. We will call this competitive ratio $\alpha^*_{n_1,m_1}$.

Proof. For bid profile m (where m agents bid 1 and the remaining agents bid 0), the perfect (omnipotent) mechanism would achieve an efficiency of m. Hence, $\frac{U_{M_{n_1,m_1}}(m)}{m}$

is an upper bound on $\alpha^*_{n_1,m_1}$.

We now show that $\frac{U_{M_{n_1,m_1}}(m)}{m}$ is equal to the second expression in the proposition; then, we will show that M_{n_1,m_1} does in fact attain this competitive ratio. In the (n_1, m_1)-partition mechanism, n_1 agents are randomly picked to form one group, and the remaining $n - n_1$ agents form a second group. If m agents bid 1 and the remaining agents bid 0, then the probability of having x agents that bid 1 in the group of size n_1 is $\frac{\binom{n_1}{x}\binom{n-n_1}{m-x}}{\binom{n}{m}}$. The corresponding total welfare is $(\min\{x, m_1\} + \min\{m - x, m - m_1\})$. The set of possible values of x is X. It follows that $U_{M_{n_1,m_1}}(m)$ is equal to $\frac{\sum_{x\in X}\binom{n_1}{x}\binom{n-n_1}{m-x}(\min\{x,m_1\}+\min\{m-x,m-m_1\})}{\binom{n}{m}}$.

All that is left to show is that M_{n_1,m_1} does in fact attain this competitive ratio. Let us consider the following allocation mechanism, which is never better than M_{n_1,m_1}:

- Pick n_1 agents to form one group, uniformly at random. The other $n - n_1$ agents form the second group.

- Remove the agents with the lowest $n - m$ bids.

- For the first group, if there are more than m_1 agents left, allocate m_1 units uniformly at random among the remaining agents in group one. Otherwise, allocate one unit to every remaining agent in group one.

- For the second group, if there are more than $m - m_1$ agents left, allocate $m - m_1$ units uniformly at random among the remaining agents in group two. Otherwise, allocate one unit to every remaining agent in group two.

For any bid profile, the above mechanism results in (weakly) lower efficiency than the (n_1, m_1)-partition mechanism, because in the partition mechanism, the units are assigned efficiently within each group, and in the modified mechanism they are not because of agent removal and random assignment.

Under the modified mechanism, only the agents bidding $v_1, \ldots, v_m$ possibly win any units, and the probability of winning is the same for each of them. For the bid profile in which m agents bid 1 and the remaining agents bid 0, the modified mechanism results in the same efficiency as the partition mechanism. Therefore, because in this case, a winning agent's utility is 1, the expected number of winners under the modified mechanism is $U_{M_{n_1,m_1}}(m)$. But this probability must be the same for all bid profiles. So, using the fact that each of the top m bidders is equally likely to win, for a general bid profile, the expected efficiency under the modified mechanism is $\frac{U_{M_{n_1,m_1}}(m)}{m} \sum_{i=1}^{m} v_i$; and we know that this is (weakly) lower than the expected efficiency under the (n_1, m_1)-partition mechanism. Hence, the (n_1, m_1)-partition mechanism has a competitive ratio of at least $\frac{U_{M_{n_1,m_1}}(m)}{m}$. We have already proved that $\frac{U_{M_{n_1,m_1}}(m)}{m}$ is an upper bound of $\alpha^*_{n_1,m_1}$, so this expression must be exactly equal to the competitive ratio. □

So far, we have not considered mixtures over partition mechanisms. It could be that, by taking such mixtures, we can obtain more competitive mechanism. However, the following proposition rules out the possibility of obtaining more competitive mechanisms by taking mixtures over partition mechanisms.

Proposition 48. *If M is a mixture over $M_1, M_2, \ldots, M_t$, where the M_i are partition mechanisms for different values of n_1, m_1, and M_i is chosen with probability p_i, then there exists $1 \leq j \leq t$ so that M_j attains at least the competitive ratio of M.*

Proof. By the same argument as in Proposition 47, the competitive ratio of M is at most $\frac{U_M(m)}{m}$. We have that $\frac{U_M(m)}{m} = \frac{\sum_{i=1}^{t} p_i U_{M_i}(m)}{m} \leq \max_j \frac{U_{M_j}(m)}{m}$. But $\frac{U_{M_j}(m)}{m}$ (where $j \in \arg\max_j \frac{U_{M_j}(m)}{m}$) is the competitive ratio for M_j by Proposition 47. Hence, M_j is as competitive as M. □

By Proposition 47, for given n and m, by maximizing

$$\frac{\sum_{x\in X}\binom{n_1}{x}\binom{n-n_1}{m-x}(\min\{x,m_1\}+\min\{m-x,m-m_1\})}{m\binom{n}{m}}$$

over n_1 and m_1, we obtain the optimal (n_1, m_1)-partition mechanism. This mechanism is also optimal among all mixtures of partition mechanisms by Proposition 48. It would be nice to have a general analytical characterization of the optimal n_1 and m_1. The following conjecture specifies three partition mechanisms, and gives the corresponding competitive ratios. The conjecture states that for any n and m, the optimal partition mechanism must be one of these three. Experimentally, we have verified that this conjecture is true for all $n \leq 10$.

Conjecture 2. *For any n and m, the optimal partition mechanism is one of the following three:*
$(1, 0)$*-partition mechanism, with competitive ratio* $\frac{n-1}{n}$*;*
$(1, 1)$*-partition mechanism, with competitive ratio* $\frac{nm+m-n}{nm}$*;*
$(2, 1)$*-partition mechanism, with competitive ratio*

$$\frac{\sum_{x\in X'}\binom{2}{x}\binom{n-2}{m-x}(\min\{x,1\}+\min\{m-x,m-1\})}{m\binom{n}{m}},$$

where $X' = \{x|0 \leq x \leq 2, 0 \leq m - x \leq n - 2\}$*.*

In Table 2.11, we present the results for various numbers of agents and units. The second column ($\alpha^*_{n_1,m_1}$) gives the optimal competitive ratio among all partition mechanisms. The third column gives the values of n_1 and m_1, where the (n_1, m_1)-partition mechanism achieves the optimal competitive ratio.

2.4.5 Generalized partition mechanisms

Finally, we slightly generalize the definition of partition mechanisms by allowing for empty groups of agents in the partition, as well as burning units.

Table 2.11: Competitive ratios of partition mechanisms.

	$\alpha^*_{n_1,m_1}$	(n_1, m_1)
$n = 4, m = 1$	0.750	$(1, 0)$
$n = 4, m = 2$	0.833	$(2, 1)$
$n = 4, m = 3$	0.917	$(1, 1)$
$n = 6, m = 1$	0.833	$(1, 0)$
$n = 6, m = 3$	0.867	$(2, 1)$
$n = 6, m = 5$	0.967	$(1, 1)$
$n = 8, m = 1$	0.875	$(1, 0)$
$n = 8, m = 3$	0.875	$(1, 0)$
$n = 8, m = 5$	0.925	$(1, 1)$
$n = 8, m = 7$	0.982	$(1, 1)$

Definition 5. Given n and m, for nonnegative integers n_1, n_2, m_1, m_2 with $n_1+n_2 = n$, $m_1 + m_2 \leq m$, we define the following mechanism:

- Pick n_1 agents to form one group, uniformly at random. The other $n - n_1$ agents form the second group. (One group can be empty.)
- Allocate m_1 units among the first group, according to the VCG mechanism.
- Allocate m_2 units among the second group, according to the VCG mechanism.

We call this mechanism the (n_1, m_1, m_2)-*generalized partition mechanism.*

We removed the transferring of VCG revenue from the definition, because when one group is empty, it is not possible to transfer to that group. However, we still allow for redistribution, so if both groups are nonempty (or, more generally, if we randomize only over generalized partition mechanisms in which both groups are nonempty) we will in fact redistribute all the VCG revenue.

The set of generalized partition mechanisms contains all the burning allocation mechanisms: the $(0, 0, m_2)$-generalized partition mechanism is the mechanism in which $m - m_2$ units are burned, and the remaining units are allocated efficiently among all agents.

Proposition 49. *All generalized partition mechanisms are strategy-proof and linear.*

Proof. Every agent is participating in a VCG mechanism, which must be strategy-proof and individually rational. We also have that if an agent's bid is 0, then her payment is 0. Let M be a generalized partition mechanism, U_M and P_M are the average of the efficiency and VCG revenue over all random partitions of the agents into groups of sizes n_1 and $n - n_1$. Given a specific way of partitioning, both the efficiency and the VCG revenue are linear in the v_i. Therefore, both U_M and P_M are linear in the v_i as well. □

We can now directly apply Theorem 9 to find the mechanism (M, R) with the highest competitive ratio, given that M is a mixture of the generalized partition mechanisms. In Table 2.12, we present the results for various numbers of agents and units. The second column (α^*) gives the optimal competitive ratio among all (M, R), under the constraint that M is a mixture of the generalized partition mechanisms. The third column describes a mixture of generalized partition mechanisms that attains the optimal competitive ratio in each case (the meaning of $(n_1, m_1, m_2), p$ is that with probability p, we use the (n_1, m_1, m_2)-generalized partition mechanism). (We do not present the redistribution function because we do not know how to conveniently describe it in a table.)

Table 2.12: Competitive ratios of generalized partition mechanisms.

	α^*	allocation mechanism
$n = 4, m = 1$	0.842	$(0, 0, 1), 0.37$ $(1, 0, 1), 0.63$
$n = 4, m = 2$	0.864	$(0, 0, 2), 0.18$ $(2, 1, 1), 0.82$
$n = 4, m = 3$	0.923	$(0, 0, 3), 0.08$ $(1, 1, 2), 0.92$
$n = 8, m = 1$	0.962	$(0, 0, 1), 0.69$ $(1, 0, 1), 0.31$
$n = 8, m = 3$	0.908	$(0, 0, 3), 0.26$ $(1, 0, 3), 0.74$
$n = 8, m = 5$	0.928	$(0, 0, 5), 0.04$ $(1, 1, 4), 0.96$
$n = 8, m = 7$	0.982	$(0, 0, 7), 0.02$ $(1, 1, 6), 0.98$

2.5 Summary

In this chapter, we applied CFAMD to the problem of designing resource allocation mechanisms that redistribute their revenue back to the agents. For allocation problems with one or more items, the well-known Vickrey-Clarke-Groves (VCG) mechanism is efficient, strategy-proof, individually rational, and does not incur a deficit. However, it is not (strongly) budget balanced: generally, the agents' payments will sum to more than 0. We studied mechanisms that redistribute some of the VCG payments back to the agents, while maintaining the desirable properties of the VCG mechanism. In Section 2.1, we focused on designing VCG redistribution mechanisms that redistribute the most in the *worst case*. For auctions with multiple indistinguishable units in which marginal values are nonincreasing, we derived a mechanism that is optimal in this sense. We also showed that if marginal values are not required to be nonincreasing, then the original VCG mechanism is worst-case optimal. In Section 2.2, we studied the problem of designing VCG redistribution mechanisms that redistribute the most in *expectation* when prior distributions over the agents' valuations are available. For auctions with multiple indistinguishable units in which each agent is only interested in one unit, we analytically derived the OEL mechanism that is optimal among linear redistribution mechanisms. For this setting, we also proposed an automated mechanism design technique based on type discretization. We then generalized our setting to auctions with multiple indistinguishable units in which marginal values are nonincreasing. We extended the notion of linear redistribution mechanisms to this more general setting. In Section 2.3, we studied the problem of designing mechanisms whose redistribution functions are *undominated* in the sense that no other mechanisms can always perform as well, and sometimes better. We introduced two measures for comparing two VCG redistribution mechanisms with respect to how well off they make the agents, and studied the question of finding maximal elements in the space of non-deficit redistribution mechanisms, with respect to the partial orders induced by both measures. One main discovery is that, for auctions with multiple indistinguishable units, where each agent is only interested in a single copy of the unit, if we restrict our attention to linear and anonymous redistribution functions, then the maximal elements defined by both measures coincide, and they are exactly the family of OEL mechanisms. Finally, in Section 2.4, we studied the problem of designing the allocation rule together with the redistribution scheme, allowing for the allocation to be inefficient. We proposed several specific mechanisms that are based on burning items, excluding agents, and (most generally) partitioning the items and agents into groups.

3

Mechanism Design Without Payments

In Chapter 2, we tried minimizing the net payments by the agents, but the payments were nevertheless a very helpful tool to create the right incentives. In this chapter, however, we study the problem of designing resource allocation mechanisms that do not rely on payments. This is useful in settings where no currency has (yet) been established (as may be the case, for example, in a peer-to-peer network, as well as in many other multiagent systems); or where payments are prohibited by law; or where payments are otherwise inconvenient. Specifically, our objective is to design mechanisms that do not rely on payments, and lead to high social welfare.

The problem of allocating resources among multiple competing agents when monetary transfers are possible has been studied extensively in both the one-shot mechanism design setting [20, 8, 94, 56, 84, 39, 81], and the repeated setting [22, 11, 21, 6, 47, 72, 73]. There is also a rich literature on mechanisms without payments. A survey is given in the book chapter by Schummer and Vohra [101]. Barberà [10] gives an introduction to strategy-proof social choice functions. Budish [16] gives a nice survey of existing allocation mechanisms without payments that are designed for practical usage (*e.g.*, the patented Adjusted Winner Procedure [14]). All these mechanisms are manipulable except for the Serial Dictatorship mechanism in a paper by Budish and Cantillon [17], in which the authors study user behavior in Harvard Business School course allocation. The recently proposed qualitative Vickrey auction [63], a generalization of the traditional Vickrey auction, is another mechanism that does not necessarily rely on monetary payments. However, it cannot be applied to our problem as it requires that there will be only a single winner, and that the center has preferences over the outcomes. Mechanism design without payments has also been studied in the contexts of matching [64] and cake-cutting [15, 23].

In this chapter, we design resource allocation mechanisms that do not rely on monetary payments in two specific settings: repeated allocation of a single item among multiple agents, and single-round allocation of multiple items between two agents. The mechanisms we propose, which are based on *artificial* payments, turn out to be competitive against the optimal mechanisms with payments (in the settings that we study, the optimal mechanisms with payments produce first-best results, *i.e,* optimal efficiency).

By proposing specific competitive mechanisms that do not rely on payments, this chapter also provides an answer to the question: *Are monetary payments necessary for designing good mechanisms?* Our results imply that, sometimes, artificial payments are "good enough" for designing allocation mechanisms with high social welfare.

The idea of designing mechanisms without payments to achieve competitive performance against mechanisms with payments was explicitly framed by Procaccia and Tennenholtz [95], in their paper titled *Approximate Mechanism Design Without Money*. That paper carries out a case study on locating a public facility for agents with single-peaked preferences. (The general idea of approximate mechanism design without payments dates back further, at least to work by Dekel *et al.* [43] in a machine learning framework.) To our knowledge, along this line of research, we are the first to study allocation of private goods. Unlike the models studied in the above two papers [43, 95], where agents may have consensus agreement, when we are considering the allocation of private goods, the agents are fundamentally in conflict.[1] Nevertheless, it turns out that even here, some positive results can be obtained. Thus, we believe that our results provide additional insights to this line of research. Of course, it is beyond the scope of this chapter to answer the above question in its general form; rather, we will be content to focus specifically on designing payment-free allocation mechanisms with high social welfare.

Most of the payment-free allocation mechanisms proposed in this chapter have been obtained with the help of the CFAMD approach. Previously, the basic AMD approach (without parameterization) has been applied to settings without payments (*e.g.*, [33]). To our knowledge, this dissertation is the first to use automated mechanism design to study parameterized families of mechanisms that do not rely on payments.

[1] For example, both [43] and [95] proposed mechanisms that pick the "median" report from the agents as the final outcome. When the agents' favorite outcomes are identical, the median report is the consensus agreement for all the agents. When allocating private goods (without externalities), consensus agreement never exists—every agent wants every good. Of course, in the worst case (all of these papers are based on worst-case analysis), the agents in the earlier papers are also in conflict.

3.1 Competitive Repeated Allocation Without Payments

In this section, we study the problem of allocating a single item repeatedly among multiple competing agents, in an environment where monetary transfers are not possible. An example scenario is an operating system that needs to allocate CPU time slots to different applications. The resource in this example is the CPU (we assume that we are dealing with single-core CPUs) and the agents are the applications. Another example scenario, closer to daily life, is "who gets the TV remote control." Here the resource is the remote control and the agents are the members of the household. In both scenarios the resource is allocated repeatedly among the agents, and monetary transfers are infeasible (or at least inconvenient). In this section, we investigate problems like the above. Our objective is to maximize social welfare, *i.e.*, allocative efficiency.[2]

We first focus on the case of two agents. We introduce an artificial payment system, which enables us to construct *repeated* allocation mechanisms *without payments* based on *one-shot* allocation mechanisms *with payments*. Under certain restrictions on the discount factor, we propose several repeated allocation mechanisms based on artificial payments. For the simple model in which the agents' valuations are either high or low, the mechanism we propose is 0.94-competitive against the optimal allocation mechanism with payments. For the general case of any prior distribution, the mechanism we propose is 0.85-competitive. We generalize the mechanism to cases of three or more agents. For any number of agents, the mechanism we obtain is at least 0.75-competitive. The obtained competitive ratios imply that for repeated allocation, artificial payments may be used to replace real monetary payments, without incurring too much loss in social welfare.

[2] In Chapter 2, social welfare and allocative efficiency were not identical because there were payments that were not collected by any agent, but in this context there are no payments so the two coincide.

A paper that lays out many of the foundations for repeated games is due to Abreu *et al.* [2], in which the authors investigate the problem of finding pure-strategy sequential equilibria of repeated games with imperfect monitoring. Their key contribution is the state-based approach for solving repeated games, where in equilibrium, the game is always in a *state* which specifies the players' long-run utilities, and on which the current period's payoffs are based. There are many papers that rely on the same or a similar state-based approach [100, 82, 66, 19, 70].

The following papers are the most related to our work: Fudenberg *et al.* [48] give a folk theorem for repeated games with imperfect public information. Both [48] and this section are built based on the (dynamic programming style) *self-generating* technique in [2] (it is called *self-decomposable* in [48]). However, [48] considers self-generating based on certain supporting hyper-plane, which is guaranteed to exist only when the discount factor goes to 1.[3] Therefore, their technique does not apply to our problem because we are dealing with non-limit discount factors.[4] Another difference between [48] and this section is that we are designing specific mechanisms, instead of trying to prove the existence of a certain class of mechanisms. With non-limit discount factors, it is generally difficult to precisely characterize the set of feasible utility vectors (optimal frontier) for the agents. Several papers have already proposed different ways of approximation (for cases of non-limit discount factors). Athey *et al.* [5] study approximation by requiring that the payoffs of the agents must be symmetric. In what, from a technical perspective, appears to be the paper closest to the work in this section, Athey and Bagwell [4] investigate collusion in a

[3] In [48], it is shown that any feasible and individually rational equilibrium payoff vector v can be achieved in a perfect public equilibrium (self-generated based on a particular supporting hyper-planes), as long as the discount factor reaches a threshold $\underline{\beta}$. However, the threshold $\underline{\beta}$ depends on v. If we consider all possible values of v, then we essentially require that the discount factor/threshold approach 1, since any discount factor that is strictly less than 1 does not work (for some v).

[4] In this section, we also require that the discount factor reaches a threshold, but here the threshold is a constant that works for all possible priors.

repeated game by approximating the optimal frontier by a line segment (the same technique also appears in the work of Abdulkadiroğlu and Bagwell [1]). One of their main results is that if the discount factor reaches a certain threshold (still strictly less than 1), then the approximation comes at no cost. That is, the optimal (first-best) performance can be obtained. However, their technique only works for finite type spaces, as it builds on uneven tie-breaking. In this section, we introduce a new technique for approximating the optimal frontier for the repeated allocation problem. Our technique works for non-limit discount factors and is not restricted to symmetric payoffs or finite type spaces. The technique we propose is presented in the form of an artificial payment system, which corresponds to approximating the optimal frontier by triangles. The artificial payment system enables us to construct repeated allocation mechanisms without payments based on one-shot allocation mechanisms with payments.

The remainder of this section is organized as follows. In Subsection 3.1.1, we describe the model that we study. In Subsection 3.1.2, we review the basic results on the standard state-based approach introduced in Abreu *et al.* [2], specifically in the context of repeated allocation. In Subsection 3.1.3, we describe a numerical solution technique based on the state-based approach. The proposed numerical solution technique is very close to the direct AMD approach. In Subsection 3.1.4, we adopt the CFAMD approach. We introduce an artificial payment system, based on which we analytically characterize several competitive mechanisms without payments for different settings. Finally, in Subsection 3.1.5, we generalize the results to cases of three or more agents.

3.1.1 Formalization

We study the problem of allocating a single item repeatedly between two (and later in the section, more than two) competing agents. We assume that the agents' preferences are independent and identically distributed, across agents as well as allocation periods, according to a distribution F. We assume that these valuations are non-negative and have finite expectations. F does not change over time. Before each allocation period, the agents learn their (private) valuations for having the item in that period (but not for any future periods). There are infinitely many periods, and agents' valuations are discounted according to a discount factor β. Our objective is to design a mechanism that maximizes expected social welfare under the following constraints (we allow randomized mechanisms):

- *(Bayes-Nash) Incentive Compatibility:* Truthful reporting is a Bayes-Nash equilibrium, that is, for agents who only care about their expected utilities, if an agent reports truthfully, then the other agent's best response is also to report truthfully.[5]
- *No Payments:* No monetary transfers are ever made.

In the one-shot mechanism design setting, incentive compatibility is usually achieved through payments. This ensures that agents have no incentive to overbid, because they may have to make large payments. In the repeated allocation setting, there are other ways to achieve incentive compatibility: for example, if an agent strongly prefers to obtain the item in the current period, the mechanism can ensure that she is less likely to obtain it in future periods. In a sense, this is an artificial form of payment. Such payments introduce some new issues that do not always occur with monetary payments, including that each agent effectively has a limited budget (corresponding to a limited amount of future utility that can be given up); and if one agent makes a payment to another agent by sacrificing some amount of future

[5] In other parts of this dissertation, we study strategy-proof mechanisms. This section is the only exception.

utility, the corresponding increase in the latter agent's utility may be different from the decrease in the former agent's utility.

3.1.2 State-Based Approach

Throughout the section, we adopt the state-based approach introduced in Abreu *et al.* [2]. In their paper, the authors investigated the problem of finding pure-strategy sequential equilibria of repeated games with imperfect monitoring. Their problem can be rephrased as follows: Given a game, what are the possible pure-strategy sequential equilibria? Even though here we are considering a different problem (we are *designing* the game), the underlying ideas still apply. In their paper, states correspond to possible equilibria, while here, states correspond to feasible mechanisms. In this subsection, we review a list of basic results and observations on the state-based approach, specifically in the context of repeated allocation.

Let M be an incentive compatible mechanism without payments for a particular (fixed) repeated allocation problem, defined by a particular type distribution and a discount factor. If, under M, the expected long-term utilities of agents 1 and 2 (at the beginning) are x and y respectively, then we denote mechanism M by state (x, y). All mechanisms that can be denoted by (x, y) are considered equivalent. If we are about to apply mechanism M, then we say the agents are in state (x, y). In the first period, based on the agents' reported values, the mechanism specifies both how to allocate the item in this period, and what to do in the future periods. The rule for the future is itself a mechanism. Hence, a mechanism specifies how to allocate the item within the first period, as well as the state (mechanism) that the agents will be in in the second period. We have that $(x, y) = E_{v_1,v_2}[(r_1(v_1, v_2), r_2(v_1, v_2)) + \beta(s_1(v_1, v_2), s_2(v_1, v_2))]$, where v_1, v_2 are the first-period valuations, r_1, r_2 are the immediate rewards obtained from the first-period *allocation rule*, and (s_1, s_2) gives the second-period state, representing the *transition rule*.

State (x, y) is called a *feasible* state if there is a feasible mechanism (that is, an incentive compatible mechanism without payments) corresponding to it. We denote the set of feasible states by S^*. Let e be an agent's expected valuation for the item in a single period. $E = \frac{e}{1-\beta}$ is the maximal expected long-term utility an agent can receive (corresponding to the case where she receives the item in every period). Let O be the set of states $\{(x, y)|0 \leq x \leq E, 0 \leq y \leq E\}$. We have that $S^* \subseteq O - \{(E, E)\} \subsetneq O$.

S^* is convex, for the following reason. If (x_1, y_1) and (x_2, y_2) are both feasible, then $(\frac{x_1+x_2}{2}, \frac{y_1+y_2}{2})$ is also feasible (it corresponds to the randomized mechanism where we flip a coin to decide which of the two mechanisms to apply). S^* is symmetric with respect to the diagonal $y = x$: if (x, y) is feasible, then so is (y, x) (by switching the roles of the two agents).

The approximate shape of S^* is illustrated in Figure 3.1. There are three noticeable extreme states: $(0, 0)$ (nobody ever gets anything), $(E, 0)$ (agent 1 always gets the item), and $(0, E)$ (agent 2 always gets the item). S^* is confined by the x-axis (from $(0, 0)$ to $(E, 0)$), the y-axis (from $(0, 0)$ to $(0, E)$), and, most importantly, the bold curve, which corresponds to the optimal frontier. The square specified by the dotted lines represents O.

Our objective is to find the state $(x^*, y^*) \in S^*$ that maximizes $x^* + y^*$ (expected social welfare). By convexity and symmetry, it does not hurt to consider only cases where $x^* = y^*$.

We now define a notion of when one set of states is *generated by* another. Recall that a mechanism specifies how to allocate the item within the first period, as well as which state the agents transition to for the second period. Let S be any set of states with $S \subset O$. Let us assume that, in the second period, exactly the states in S are feasible. That is, we assume that, if and only if $(x, y) \in S$, starting at the second

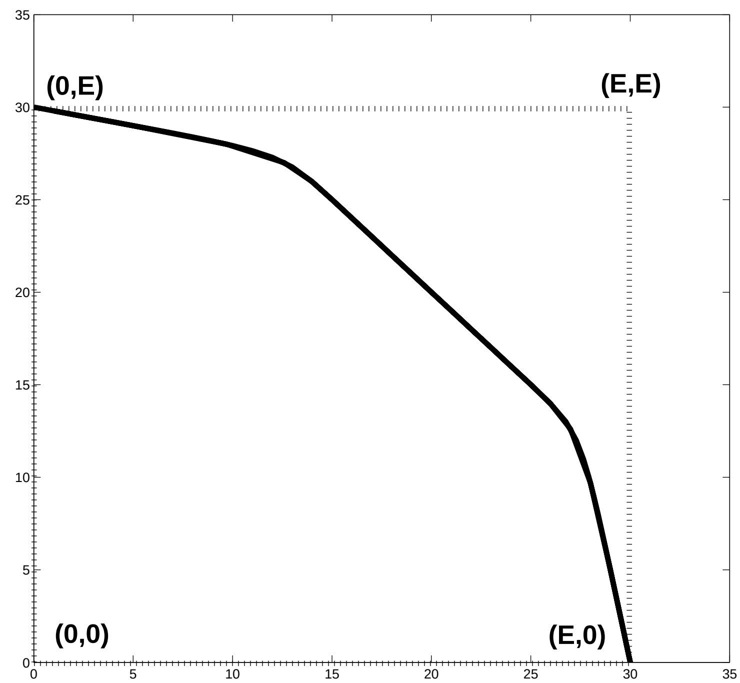

FIGURE 3.1: An illustration of the set of all feasible states (mechanisms) for repeated allocation without payments.

period, there exists a feasible mechanism under which the expected utilities of agent 1 and 2 are x and y, respectively. Based on this assumption, we can construct incentive compatible mechanisms starting at the first period, by specifying an *allocation rule* for the first period, as well as a *transition rule* that specifies the states in S to which the agents will transition for the beginning of the second period. Now, we only need to make sure that the first period is incentive compatible. That is, the allocation rule in the first period, combined with the rule for selecting the state at the start of the second period, must incentivize the agents to report their true valuations in the first period. We say the set of resulting feasible states for the first period is *generated by* S, and is denoted by $Gen(S)$.

The following proposition provides a general guideline for designing feasible mechanisms.

Proposition 50. *For any $S \subseteq O$, if $S \subseteq Gen(S)$, then $S \subseteq S^*$. That is, if S is* self-generating*, then all the states in S are feasible.*

Proof. Let (x, y) be any state in S. (x, y) is also in $Gen(S)$. Let $M(x, y)$ be the way in which (x, y) is generated. That is, $M(x, y)$ specifies an allocation rule within the first period, as well as a transition rule that determines the state (x', y') to which agents will transition for the start of the second period. Because any such (x', y') is in S, it is also in $Gen(S)$, so there exists some $M(x', y')$ that generates it, which we can then use in the next period—*etc.* This defines a complete mechanism; moreover, this is incentive compatible because the combination of an allocation and a transition rule in each period is incentive compatible. Hence, (x, y) is indeed in S^*. □

It should be noted that in the proof of Proposition 50, we used the fact that F (prior distribution) does not change over time. That is, if F is not fixed, then the above proposition may not hold. The above proof also provides a guideline for constructing mechanisms that correspond to a given state. In general, there are many mechanisms corresponding to a given state (for continuous F, any two mechanisms that differ on a measure zero set of bid profiles correspond to the same state).

We now consider starting with the square O that contains S^* and iteratively generating sets. Let $O^0 = O$ and $O^{i+1} = Gen(O^i)$ for all i. The following proposition, together with Proposition 50, provide a general approach for computing S^*.

Proposition 51. *$S^* = Gen(S^*)$.*

Proof. A state is feasible if and only if it can be generated from feasible states, and S^* consists exactly of all feasible states. □

Proposition 52. *For any i, we have $S^* \subseteq O^i$.*

Proof. We know $S^* \subseteq O^0$. If $S \subseteq S'$, then $Gen(S) \subseteq Gen(S')$. By Proposition 51, $Gen(S^*) = S^*$, so if $S^* \subseteq O^i$, then $S^* = Gen(S^*) \subseteq Gen(O^i) = O^{i+1}$. Hence, the proposition follows by induction. □

Proposition 53. *The O^i form a sequence of (weakly) decreasing sets That is, $O^{i+1} \subseteq O^i$ for all i.*

Proof. We have $O^1 \subseteq O^0$, because even if we assume that all the states in O are feasible starting in the second period, an agent's expected utility is still between 0 and E. Moreover, if $O^{i+1} \subseteq O^i$, then $O^{i+2} = Gen(O^{i+1}) \subseteq Gen(O^i) = O^{i+1}$. Hence, the proposition follows by induction. □

Proposition 54. *If $O^i = O^{i+1}$, then $O^i = S^*$. That is, the O^i form a sequence that converges to S^* if it converges at all.*

Proof. If $O^i = O^{i+1}$, then by Proposition 50, $O^i \subseteq S^*$. By Proposition 52, $S^* \subseteq O^i$. Therefore, we have $O^i = S^*$. □

The above guideline leads to a numerical solution technique for finite type spaces. With a properly chosen numerical discretization scheme, we are able to compute an underestimation of O^i for all i, by solving a series of linear programs. The underestimations of the O^i always converge to an underestimation of S^* (a subset of S^*). That is, we end up with a set of feasible mechanisms. We are also able to show that as the discretization step size goes to 0, the obtained feasible set approaches S^*. That is, the numerical solution technique produces an optimal mechanism in the limit as the discretization becomes finer. Details of the numerical solution technique are presented in the next subsection.

3.1.3 Numerical Solution

Here we provide more details of the numerical solution technique. The technique can be applied to any finite type space. The technique solves for a mechanism that

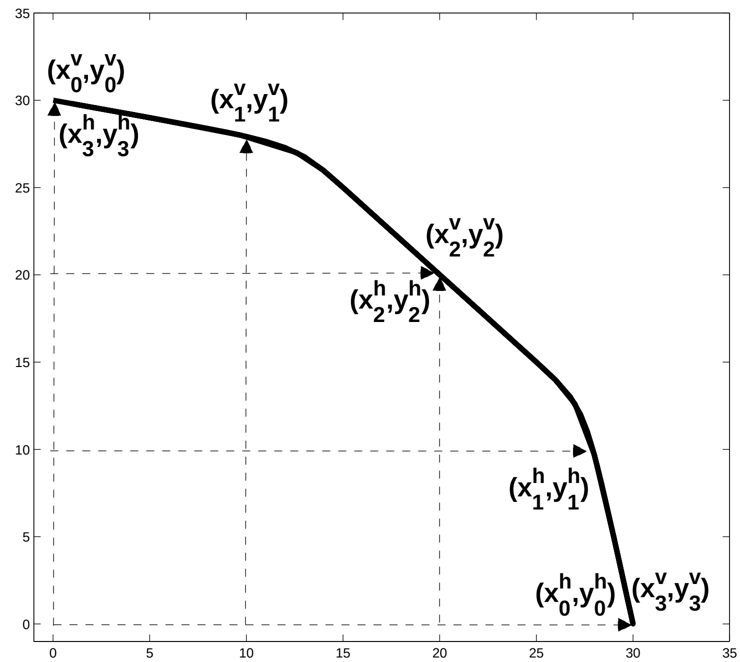

FIGURE 3.2: Discretization scheme for numerically solving for optimal repeated allocation mechanisms without payments.

is very close to optimal. As the discretization step size goes to 0, the technique produces an optimal mechanism in the limit.

The O^i as defined in Subsection 3.1.2 all have similar shapes to S^*. They are all convex (because any set $Gen(S)$ is). They all contain $(0,0)$, $(E,0)$, and $(0,E)$. The areas are all confined by the x-axis (from $(0,0)$ to $(E,0)$), the y-axis (from $(0,0)$ to $(0,E)$), and a curve from $(0,E)$ to $(E,0)$. Sets with the above shape are the only sets that we will ever come across in this subsection. Let S be one such set. We will use convex combinations of the following states to approximate S (illustrated in Figure 3.2):

- For $i = 0, 1, \ldots, N$, we have state (x_i^v, y_i^v), where $x_i^v = ih$ and y_i^v is the largest value satisfying $(x_i^v, y_i^v) \in S$.

- For $i = 0, 1, \ldots, N$, we have state (x_i^h, y_i^h), where $y_i^h = ih$ and x_i^h is the largest value satisfying $(x_i^h, y_i^h) \in S$.

- $(0, 0)$, $(0, E)$, and $(E, 0)$.

Here, h is the step size; $h = E/N$. The x_i^v and the y_i^h are fixed. Thus in our representation, S is characterized by the y_i^v and the x_i^h.

We denote the above $k = 2N + 5$ states by (x_i, y_i) for $i = 1, 2, \ldots, k$. (Some of these states are identical.)

Let $D(S)$ be the set of states that are convex combinations of the above k states. We have $D(S) \subseteq S$, because S is convex, and all the above k states are in S. In general, $D(S) \subsetneq S$, because some states in S may be missing as a result of the discretization (those that are not convex combinations of the above k states). As the step size h gets smaller, fewer states are missing.

We denote the (finite) type space by $\{\theta_1, \theta_2, \ldots, \theta_n\}$. Here, θ_i represents one allowable type in the finite type space Ω, instead of agent i's type (the usual definition of θ_i). The prior distribution F assigns probability p_i to value θ_i. $\sum_i p_i = 1$.

Any mechanism that is generated by $D(S)$ can be described by the following variables:

- a_{ij}^1: The probability that the item is allocated to player 1 in the first period when 1 reports θ_i and 2 reports θ_j.

- a_{ij}^2: The probability that the item is allocated to player 2 in the first period when 1 reports θ_i and 2 reports θ_j.

- (x_{ij}, y_{ij}): The state the agents will go to at the beginning of the second period, when 1 reports θ_i and 2 reports θ_j.

- c_{ijt}: The coefficients of the convex combination to obtain that state. $x_{ij} = \sum_{t=1}^{k} c_{ijt} x_t$ and $y_{ij} = \sum_{t=1}^{k} c_{ijt} y_t$. Here, the k states (x_i, y_i) $(i = 1, 2, \ldots, k)$ are the states that define (vertices of) $D(S)$.

The above variables must satisfy the following constraints:

- The item should not be allocated more than once.

  For all i and j, $a_{ij}^1 + a_{ij}^2 \leq 1$.

- The future state must be in $D(S)$.

  For all i and j, $x_{ij} = \sum_{t=1}^{k} c_{ijt} x_t$ and $y_{ij} = \sum_{t=1}^{k} c_{ijt} y_t$.

  For all i, j, and t, $c_{ijt} \geq 0$.

  For all i and j, $\sum_{t=1}^{k} c_{ijt} = 1$.

- (Bayes-Nash) incentive compatibility must be satisfied.

  For all i and i', $\theta_i(\sum_{j=1}^{n} p_j a_{ij}^1) + \beta(\sum_{j=1}^{n} p_j x_{ij})$
  $\geq \theta_i(\sum_{j=1}^{n} p_j a_{i'j}^1) + \beta(\sum_{j=1}^{n} p_j x_{i'j})$.

  For all j and j', $\theta_j(\sum_{i=1}^{n} p_i a_{ij}^2) + \beta(\sum_{i=1}^{n} p_i y_{ij})$
  $\geq \theta_j(\sum_{i=1}^{n} p_i a_{ij'}^2) + \beta(\sum_{i=1}^{n} p_i y_{ij'})$.

This gives the set $\hat{S} = Gen(D(S))$ of states (x, y), satisfying:

$x = \sum_{i=1}^{n} p_i(\theta_i(\sum_{j=1}^{n} p_j a_{ij}^1) + \beta(\sum_{j=1}^{n} p_j x_{ij}))$

$y = \sum_{j=1}^{n} p_j(\theta_j(\sum_{i=1}^{n} p_i a_{ij}^2) + \beta(\sum_{i=1}^{n} p_i y_{ij}))$

where the variables in these equations must satisfy the above constraints.

To approximately represent $\hat{S}$, we have the following k states, whose convex combination is $D(\hat{S})$.

- For $i = 0, 1, \ldots, N$, we have state $(\hat{x}_i^v, \hat{y}_i^v)$, where $\hat{x}_i^v = ih$ and $\hat{y}_i^v$ is the largest value so that $(\hat{x}_i^v, \hat{y}_i^v) \in \hat{S}$. In other words, $\hat{y}_q^v$ is the solution of the following linear program for $q = 0, 1, \ldots, N$:

Variable: $\hat{y}_q^v$, a_{ij}^1, a_{ij}^2, x_{ij}, y_{ij}, c_{ijt}
Maximize $\hat{y}_q^v$
Subject to:
$a_{ij}^1 + a_{ij}^2 \leq 1$, for all i and j.
$x_{ij} = \sum_{t=1}^k c_{ijt}x_t$, $y_{ij} = \sum_{t=1}^k c_{ijt}y_t$, for all i and j.
$c_{ijt} \geq 0$, for all i, j and t.
$\sum_{t=1}^k c_{ijt} = 1$, for all i and j.
$\theta_i(\sum_{j=1}^n p_j a_{ij}^1) + \beta(\sum_{j=1}^n p_j x_{ij}) \geq \theta_i(\sum_{j=1}^n p_j a_{i'j}^1) + \beta(\sum_{j=1}^n p_j x_{i'j})$, for all i and i'.
$\theta_j(\sum_{i=1}^n p_i a_{ij}^2) + \beta(\sum_{i=1}^n p_i y_{ij}) \geq \theta_j(\sum_{i=1}^n p_i a_{ij'}^2) + \beta(\sum_{i=1}^n p_i y_{ij'})$, for all j and j'.
$qh = \sum_{i=1}^n p_i(\theta_i(\sum_{j=1}^n p_j a_{ij}^1) + \beta(\sum_{j=1}^n p_j x_{ij}))$.
$\hat{y}_q^v = \sum_{j=1}^n p_j(\theta_j(\sum_{i=1}^n p_i a_{ij}^2) + \beta(\sum_{i=1}^n p_i y_{ij}))$.

- For $i = 0, 1, \ldots, N$, we have state $(\hat{x}_i^h, \hat{y}_i^h)$, where $\hat{y}_i^h = ih$ and $\hat{x}_i^h$ is the largest value so that $(\hat{x}_i^h, \hat{y}_i^h) \in \hat{S}$. The linear programs for computing the $\hat{x}_q^h$ are similar to those given above for the $\hat{y}_q^v$.

- $(0, 0)$, $(0, E)$, and $(E, 0)$.

For any S, the above linear programs solve for $D(\hat{S}) = D(Gen(D(S)))$. Let $\hat{O}^0 = D(O) = O$ and let $\hat{O}^{i+1} = D(Gen(D(\hat{O}^i)))$ for all i. If we start from $\hat{O}^0$, then by solving the above linear programs iteratively, we get $\hat{O}^i$ for all i.

It is straightforward to check that for any i, $\hat{O}^{i+1} \subseteq \hat{O}^i$, by a similar argument as that in the proof of Proposition 53. For any i, $\hat{O}^i$ is characterized by the y_j^v and the x_j^h for $j = 0, 1, \ldots, N$. The y_j^v and the x_j^h are nonincreasing as i increases. Hence, the $\hat{O}^i$ always converge, and the limit set is self-generating. According to Proposition 50, states in the resulting set are all feasible. Among all the feasible states obtained, we pick the state (x^*, y^*) that maximizes $x^* + y^*$ as our final solution. The above linear programs can also be used to compute the actual mechanisms corresponding to (x^*, y^*).

Proposition 55. *As the step size h goes to* 0*, the above numerical solution technique produces an optimal mechanism in the limit.*

Proof. Recall that according to the prior distribution F, an agent's valuation for the item is θ_i with probability p_i for $i = 1, 2, \ldots, n$. Let uF ($u > 1$) be the prior distribution over magnified values in which an agent's valuation is $u\theta_i$ with probability p_i for all i.

It is easy to see that the welfare achieved by the above technique under the modified prior distribution uF is just u times the welfare achieved under the original prior distribution F, if we keep the number of points in the discretization (corresponding to N) fixed.

We know that $S^* \subseteq Gen(S^*)$, and larger u corresponds to larger $Gen(S^*)$. However, $D(Gen(S^*))$ drops some states from $Gen(S^*)$ as a result of the discretization. For sufficiently large u, it will be the case that $S^* \subset D(Gen(S^*))$. Let u^* be the smallest value of u for which this is the case. As the step size h goes to 0, fewer states are dropped as a result of the discretization, and as a result, the value of u^* goes to 1 as h goes to 0.

When we apply the technique under the modified prior distribution u^*F, all the $\hat{O}^i$ contain S^*, for the following reason. Certainly, we have $S^* \subseteq \hat{O}^0$. We know that $S^* \subseteq D(Gen(S^*))$. Therefore, if $S^* \subseteq \hat{O}^i$, then $S^* \subseteq D(Gen(S^*)) \subseteq D(Gen(\hat{O}^i)) = D(Gen(D(\hat{O}^i))) = \hat{O}^{i+1}$. By induction, the $\hat{O}^i$ all contain S^*. This implies that the optimal welfare under F is less than or equal to the welfare achieved by the technique under u^*F. Hence, the welfare achieved by the technique under F is at least $\frac{1}{u^*}$ times the optimal welfare. As h goes to 0, u^* goes to 1, which means that the technique produces an optimal mechanism in the limit. □

One drawback of the numerical approach (and direct AMD approach in general) is that the obtained mechanism does not have an elegant form. This makes it harder to analyze. From the agents' perspective, it is difficult to comprehend what the mechanism is trying to do, which may lead to irrational behavior. Another drawback of the numerical approach is that it only applies to cases of finite type spaces. In what follows, we take a more analytical approach. We aim to design mechanisms that can be more simply and elegantly described, work for any type space, and are (hopefully) close to optimality.

Later on, we will compare the performances of the mechanisms obtained numerically and the mechanisms obtained by the analytical approach.

3.1.4 Competitive Analytical Mechanism

In this subsection, we propose the idea of an artificial payment system. Based on this, we propose several mechanisms that can be elegantly described, and we can prove that these mechanisms are close to optimality.

Let us recall the approximate shape of S^* (Figure 3.3). The area covered by S^* consists of two parts. The lower left part is a triangle whose vertices are $(0, 0)$, $(E, 0)$, and $(0, E)$. These three states are always feasible, and so are their convex combinations. The upper right part is a bow shape confined by the straight line and the bow curve from $(0, E)$ to $(E, 0)$. To solve for S^*, we are essentially solving for the largest bow shape satisfying that the union of the bow shape and the lower-left triangle is self-generating. Here, we consider an easier problem. Instead of solving for the largest bow shape, we solve for the largest triangle (whose vertices are $(0, E)$,$(E, 0)$, and (x^*, x^*)) so that the union of the two triangles is self-generating (illustrated in Figure 3.3). That is, we want to find the largest value of x^* that satisfies that the set of convex combinations of $(0, 0)$, $(E, 0)$, $(0, E)$, and (x^*, x^*) is self-generating.

Using the language of computationally feasible automated mechanism design, what we are doing is essentially focusing on a parameterized subfamily of mechanisms, where each mechanism within the subfamily corresponds to a certain value of x^*. Then, to optimize within the subfamily, we are essentially looking for the largest

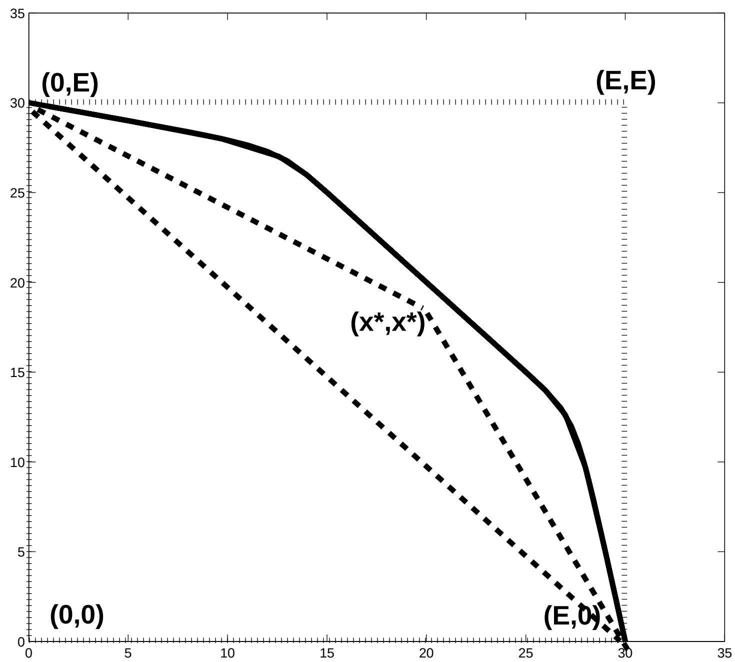

FIGURE 3.3: Triangular approximation of the set of all feasible states (mechanisms) for repeated allocation without payments.

x^*, so that (x^*, x^*) is feasible.

The triangle approximation corresponds to an *artificial payment system*. Let (x^*, x^*) be any feasible state satisfying $x^* \geq \frac{E}{2}$. Such a feasible state always exists (*e.g.*, $(\frac{E}{2}, \frac{E}{2})$). We can implement an artificial payment system based on (x^*, x^*), $(E, 0)$, and $(0, E)$, as follows. At the beginning of a period, the agents are told that the default option is that they move to state (x^*, x^*) at the beginning of the next period. However, if agent 1 wishes to pay v_1 ($v_1 \leq \beta x^*$) units of artificial currency to agent 2 (and agent 2 is not paying), then the agents will move to $(x^* - \frac{v_1}{\beta}, x^* + \frac{E-x^*}{x^*}\frac{v_1}{\beta})$. That is, the future state is moved $\frac{v_1}{\beta}$ units to the left along the straight line connecting $(0, E)$ and (x^*, x^*). (This corresponds to going to each of these two states with a certain probability.) By paying v_1 units of artificial currency, agent 1's expected utility is decreased by v_1 (the expected utility is decreased by $\frac{v_1}{\beta}$ at the start of the next period). When agent 1 pays v_1 units of artificial currency, agent 2 receives only $\frac{E-x^*}{x^*}v_1$ (also as a result of future utility). In effect, a fraction of the payment is lost in transmission. Similarly, if agent 2 wishes to pay v_2 ($v_2 \leq \beta x^*$) units of artificial currency to agent 1 (and agent 1 is not paying), then the agents will move to $(x^* + \frac{E-x^*}{x^*}\frac{v_2}{\beta}, x^* - \frac{v_2}{\beta})$. That is, the future state is moved $\frac{v_2}{\beta}$ units towards the bottom along the straight line connecting (x^*, x^*) and $(E, 0)$. If both agents wish to pay, then the agents will move to $(x^* - \frac{v_1}{\beta} + \frac{E-x^*}{x^*}\frac{v_2}{\beta}, x^* - \frac{v_2}{\beta} + \frac{E-x^*}{x^*}\frac{v_1}{\beta})$, which is a convex combination of $(0,0)$, $(0,E)$, $(E,0)$, and (x^*, x^*). Effectively, both agents have a *budget* of βx^*, and when an agent pays the other agent, there is a *gift tax* with rate $1 - \frac{E-x^*}{x^*}$.

Based on the above artificial payment system, our approach is to design repeated allocation mechanisms without payments, based on one-shot allocation mechanisms with payments. In order for this to work, the one-shot allocation mechanisms need to take the gift tax into account, and an agent's payment should never exceed the budget limit.

The budget constraint is difficult from a mechanism design perspective. We circumvent this based on the following observation. An agent's budget is at least $\beta\frac{E}{2} = \frac{e\beta}{2-2\beta}$, which goes to infinity as β goes to 1. As a result, for sufficiently large discount factors, the budget constraint will not be binding. From now on, we ignore the budget limit when we design the mechanisms. Then, for each obtained mechanism, we specify how large the discount factor has to be for the mechanism to be well defined (that is, for the budget constraint to not be violated). This allows us to work around the budget constraint. The drawback is obvious: our proposed mechanisms only work for discount factors reaching a (constant) threshold (though it is not as restrictive as studying the limit case as $\beta \to 1$).

High/Low Types

We start with the simple model in which the agents' valuations are either H (high) with probability p or L (low) with probability $1-p$. Without loss of generality, we assume that $L = 1$. We will construct a repeated allocation mechanism without payments based on the following *pay-only* one-shot allocation mechanism:

Allocation: If the reported types are the same, we determine the winner by flipping a (fair) coin. If one agent's reported value is high and the other agent's reported value is low, then we allocate the item to the agent reporting high.

Payment: An agent pays 0 if its reported type is low. An agent pays $\frac{1}{2}$ if its reported type is high (whether she wins or not); this payment does not go to the other agent.

Proposition 56. *The mechanism defined above is (Bayes-Nash) incentive compatible.*

Proof. If an agent's valuation is high and she reports high, she wins with probability $1-\frac{p}{2}$, and pays $\frac{1}{2}$. Her utility is then $H(1-\frac{p}{2})-\frac{1}{2}$. If an agent's valuation is high and she reports low, she wins with probability $\frac{1}{2}-\frac{p}{2}$, and pays 0. Her utility is then $H(\frac{1}{2}-\frac{p}{2})$, which is smaller than $H(1-\frac{p}{2})-\frac{1}{2}$. Therefore, it is optimal for an agent to report high if her valuation is high.

If an agent's valuation is low and she reports low, she wins with probability $\frac{1}{2}-\frac{p}{2}$, and pays 0. Her utility is then $\frac{1}{2}-\frac{p}{2}$. If an agent's valuation is low and she reports high, she wins with probability $1-\frac{p}{2}$, and pays $\frac{1}{2}$. Her utility is also $\frac{1}{2}-\frac{p}{2}$. Therefore, it is optimal for an agent to report low if her valuation is low. □

Now we return to repeated allocation settings. Suppose (x^*, x^*) is a feasible state. That is, we have an artificial payment system with gift tax rate $1-\frac{E-x^*}{x^*}$. We apply the above one-shot mechanism, with the modifications that when an agent pays $\frac{1}{2}$, it is paying artificial currency instead of real currency, and the other agent receives $\frac{1}{2}\frac{E-x^*}{x^*}$. We note that the amount an agent receives is only based on the other agent's reported value. Therefore, the above modifications do not affect the incentives.

Under the modified mechanism, an agent's expected utility equals $\frac{T}{2}-P+P\frac{E-x^*}{x^*}+\beta x^*$. In the above expression, $T=2p(1-p)H+p^2H+(1-p)^2$ is the expected value of the higher reported value. $\frac{T}{2}$ is then the ex ante expected utility received by an agent as a result of the allocation. $P=\frac{p}{2}$ is the expected amount of artificial payment an agent pays. $P\frac{E-x^*}{x^*}$ is the expected amount of artificial payment an agent receives. βx^* is the expected future utility by default (if no payments are made).

If both agents report low, then, at the beginning of the next period, the agents go to (x^*,x^*) by default. If agent 1 reports high and agent 2 reports low, then the agents go to $(x^*-\frac{1}{2\beta},x^*+\frac{E-x^*}{2\beta x^*})$, which is a convex combination of (x^*,x^*) and $(0,E)$. If agent 1 reports low and agent 2 reports high, then the agents go to $(x^*+\frac{E-x^*}{2\beta x^*},x^*-\frac{1}{2\beta})$, which is a convex combination of (x^*,x^*) and $(E,0)$. If both agents report high, then the agents go to $(x^*-\frac{1}{2\beta}+\frac{E-x^*}{2\beta x^*},x^*-\frac{1}{2\beta}+\frac{E-x^*}{2\beta x^*})$, which is a convex combination of (x^*,x^*) and $(0,0)$. Let S be the set of all convex combinations of $(0,0)$, $(E,0)$, $(0,E)$, and (x^*,x^*). The future states given by the above mechanism are always in S. If an agent's expected utility under this mechanism is greater than or equal to x^*, then S is self-generating. That is, (x^*,x^*) is feasible as long as x^* satisfies $x^*\leq\frac{T}{2}-P+P\frac{E-x^*}{x^*}+\beta x^*$.

We rewrite it as $ax^{*2}+bx^*+c\leq 0$, where $a=1-\beta$, $b=2P-\frac{T}{2}$, and $c=-EP$. The largest x^* satisfying the above inequality is simply the larger solution of $ax^{*2}+bx^*+c=0$, which is $\frac{\frac{T}{2}-2P+\sqrt{(2P-\frac{T}{2})^2+4(1-\beta)EP}}{2(1-\beta)}$.

This leads to a feasible mechanism M^* (corresponding to state (x^*,x^*)). The expected social welfare under M^* is $2x^*$, where x^* equals the above solution.

Mechanism M^* works as follows:

- We start at state (x^*, x^*) in the first period.

$$x^* = \frac{\frac{T}{2} - 2P + \sqrt{(2P - \frac{T}{2})^2 + 4(1-\beta)EP}}{2(1-\beta)}$$

In the above expression, $T = 2p(1-p)H + p^2H + (1-p)^2$ is the expected value of the higher reported value. $P = \frac{p}{2}$ is the expected amount of artificial payment an agent pays.

- If both agents report low, we flip a (fair) coin to determine the winner of this period. The next period still starts at (x^*, x^*).

- If agent 1 reports high and agent 2 reports low, we allocate the item to agent 1. The next period starts at $(x^* - \frac{1}{2\beta}, x^* + \frac{E-x^*}{2\beta x^*})$. That is, with probability $1 - \frac{1}{2\beta x^*}$, the next period starts at (x^*, x^*), and with probability $\frac{1}{2\beta x^*}$, the next period starts at $(0, E)$.

- Similarly, if agent 1 reports low and agent 2 reports high, we allocate the item to agent 2. The next period starts at $(x^* + \frac{E-x^*}{2\beta x^*}, x^* - \frac{1}{2\beta})$. That is, with probability $1 - \frac{1}{2\beta x^*}$, the next period starts at (x^*, x^*), and with probability $\frac{1}{2\beta x^*}$, the next period starts at $(E, 0)$.

- If both agents report high, we flip a (fair) coin to determine the winner of this period. The next period starts at $(x^* - \frac{1}{2\beta} + \frac{E-x^*}{2\beta x^*}, x^* - \frac{1}{2\beta} + \frac{E-x^*}{2\beta x^*})$. That is, with probability $1 - \frac{1}{2\beta x^*} + \frac{E-x^*}{2\beta x^{*2}}$, the next period starts at (x^*, x^*), and with probability $\frac{1}{2\beta x^*} - \frac{E-x^*}{2\beta x^{*2}}$, the next period starts at $(0, 0)$.

- If a period starts at (x^*, x^*), then we repeat the above process.

- If a period starts at $(E, 0)$, then we allocate all the items to agent 1 from this period on.

- If a period starts at $(0, E)$, then we allocate all the items to agent 2 from this period on.

- If a period starts at $(0, 0)$, then we end the mechanism.

So far, we have not considered the budget limit. For the above M^* to be well-defined (satisfying the budget constraint), we need $\beta x^* \geq \frac{1}{2}$. Since $x^* \geq \frac{E}{2} = \frac{e}{2-2\beta} \geq \frac{1}{2-2\beta}$, we only need to make sure that $\frac{\beta}{2-2\beta} \geq \frac{1}{2}$. Therefore, if $\beta \geq \frac{1}{2}$, then M^* is well-defined. For specific priors, M^* could be well-defined even for smaller β.

Next, we show that (whenever M^* is well-defined) M^* is very close to optimality. Consider the *first-best allocation mechanism*: the mechanism that always successfully identifies the agent with the higher valuation and allocates the item to this agent (for free). This mechanism is not incentive compatible, and hence not feasible. The expected social welfare achieved by the first-best allocation mechanism is $\frac{T}{1-\beta}$, which is an upper bound on the expected social welfare that can be achieved by any mechanism with (or without) payments (it is a strict upper bound, as the dAGVA mechanism [41] is efficient, incentive compatible, and budget balanced).

Definition 6. When the agents' valuations are either high or low, the prior distribution over the agents' valuations is completely characterized by the values of H and p. Let W be the expected social welfare under a feasible mechanism M. Let W^F be the expected social welfare under the first-best allocation mechanism. If $W \geq \alpha W^F$ for all H and p, then we say M is α-competitive. We call α a *competitive ratio* of M.

Proposition 57. *Whenever M^* is well-defined for all H and p, (*e.g.*, $\beta \geq \frac{1}{2}$), M^* is* 0.94*-competitive.*

Proof. The expected social welfare achieved by M^* equals $2\frac{\frac{T}{2}-2P+\sqrt{(2P-\frac{T}{2})^2+4(1-\beta)EP}}{2(1-\beta)}$. We divide this expression by the expected social welfare achieved by the first-best mechanism, resulting in the ratio $\frac{\frac{T}{2}-2P+\sqrt{(2P-\frac{T}{2})^2+4eP}}{T}$.

Table 3.1: Social welfare under different repeated allocation mechanisms that do not rely on payments.

	M^*	Optimal	First-best	Lottery
$H = 2, p = 0.2, \beta = 0.5$	2.6457	2.6725	2.7200	2.4000
$H = 4, p = 0.4, \beta = 0.5$	5.5162	5.7765	5.8400	4.4000
$H = 16, p = 0.8, \beta = 0.5$	30.3421	30.8000	30.8000	26.0000
$H = 2, p = 0.2, \beta = 0.8$	6.6143	6.7966	6.8000	6.0000
$H = 2, p = 0.8, \beta = 0.8$	9.4329	9.8000	9.8000	9.0000
$H = 16, p = 0.8, \beta = 0.8$	75.8552	77.0000	77.0000	65.0000

We know that $P = \frac{p}{2}$. Let $x = p/T$ (here, $0 \leq x \leq 1$). We may rewrite the above ratio as $\frac{1}{2} - x + \sqrt{(x - \frac{1}{2})^2 + \frac{2x(p+x-px)}{p(2-p)}}$. When x is close to 0 or close to 1, the above ratio is close to 1. When p is close to 0 or close to 1, the above ratio is close to 1 (when p is close to 0, x is also close to 0). After algebraic manipulation (with the help of symbolic computation software), we find that the other critical points (not on the boundary) simultaneously satisfy $p^4 - 8p + 4 = 0$ and $x = \frac{p^2}{p^2-2p+2}$. The exact roots of $p^4 - 8p + 4 = 0$ (quartic equation) can be found using Ferrari's formula. From this, we obtain that the minimal value of the above ratio equals 0.94 (the exact expression is too long to write down). □

As a comparison, the lottery mechanism that always chooses the winner by flipping a fair coin has competitive ratio (exactly) 0.5 (if H is much larger than L and unlikely to occur).

In Table 3.1, for different values of H, p, and β, we compare M^* to the near-optimal *feasible* mechanism obtained with the numerical solution technique. The table elements are the expected social welfare under M^*, the near-optimal feasible mechanism, the first-best allocation mechanism, and the lottery mechanism.

General Valuation Space

Here, we generalize the earlier approach to general type spaces. We let f denote the probability density function of the prior distribution. (A discrete prior distribution can always be smoothed to a continuous distribution that is arbitrarily close.)

We will construct a repeated allocation mechanism without payments based on the following *pay-only* one-shot allocation mechanism:

Allocation: The agent with the higher reported value wins the item.

Payment: An agent pays $\int_0^v tf(t)dt$ if it reports v.

This mechanism is actually a[6] dAGVA mechanism [41], which is known to be (Bayes-Nash) incentive compatible.

The process is similar to the process on the high/low type space. At the end, we obtain a feasible mechanism M^*. The expected social welfare under M^* is $2x^*$, where x^* equals $\frac{\frac{T}{2}-2P+\sqrt{(2P-\frac{T}{2})^2+4(1-\beta)EP}}{2(1-\beta)}$. Here, $T = \int_0^\infty \int_0^\infty \max\{t, v\} f(t)f(v)dtdv$ is the expected value of the higher valuation. $P = \int_0^\infty \int_0^v tf(t)dtf(v)dv$ is the expected amount an agent pays.

Mechanism M^* works as follows:

- We start at state (x^*, x^*) in the first period.

$$x^* = \frac{\frac{T}{2}-2P+\sqrt{(2P-\frac{T}{2})^2+4(1-\beta)EP}}{2(1-\beta)}$$

  In the above expression, $T = \int_0^\infty \int_0^\infty \max\{t, v\} f(t)f(v)dtdv$ is the expected value of the higher valuation. $P = \int_0^\infty \int_0^v tf(t)dtf(v)dv$ is the expected amount an agent pays.

[6] "The" dAGVA mechanism often refers to a specific mechanism in a class of Bayes-Nash incentive compatible mechanisms, namely one that satisfies budget balance. Here, we will use "dAGVA mechanisms" to refer to the entire class, including ones that are not budget-balanced. Specifically, we will only use dAGVA mechanisms in which payments are always nonnegative.

- The item is allocated to the agent with the higher reported value. The next period starts at

  $(x^* - \frac{1}{\beta}\int_0^{v_1} tf(t)dt + \frac{E-x^*}{\beta x^*}\int_0^{v_2} tf(t)dt, x^* - \frac{1}{\beta}\int_0^{v_2} tf(t)dt + \frac{E-x^*}{\beta x^*}\int_0^{v_1} tf(t)dt),$

  where v_1 and v_2 are the reported types of the two agents. This state is a convex combination of $(0,0)$, $(E,0)$, $(0,E)$, and (x^*,x^*), corresponding to a randomization over these states that we explain below. We first consider the case where $v_1 \geq v_2$.

  With probability $\frac{\int_0^{v_2} tf(t)dt}{\beta x^*} - \frac{(E-x^*)(\int_0^{v_2} tf(t)dt)}{\beta x^{*2}}$, the next period starts at $(0,0)$.

  With probability $\frac{\int_{v_2}^{v_1} tf(t)dt}{\beta x^*}$, the next period starts at $(0,E)$.

  With probability $1 - \frac{\int_0^{v_1} tf(t)dt}{\beta x^*} + \frac{(E-x^*)(\int_0^{v_2} tf(t)dt)}{\beta x^{*2}}$, the next period starts at (x^*,x^*).

  Next, we consider the case where $v_1 \leq v_2$.

  With probability $\frac{\int_0^{v_1} tf(t)dt}{\beta x^*} - \frac{(E-x^*)(\int_0^{v_1} tf(t)dt)}{\beta x^{*2}}$, the next period starts at $(0,0)$.

  With probability $\frac{\int_{v_1}^{v_2} tf(t)dt}{\beta x^*}$, the next period starts at $(E,0)$.

  With probability $1 - \frac{\int_0^{v_2} tf(t)dt}{\beta x^*} + \frac{(E-x^*)(\int_0^{v_1} tf(t)dt)}{\beta x^{*2}}$, the next period starts at (x^*,x^*).

- If a period starts at (x^*,x^*), then we repeat the above process.

- If a period starts at $(E,0)$, then we allocate all the items to agent 1 from this period on.

- If a period starts at $(0,E)$, then we allocate all the items to agent 2 from this period on.

- If a period starts at $(0,0)$, then we end the mechanism.

For the above M^* to be well-defined, we need the budget βx^* to be greater than or equal to $\int_0^\infty tf(t)dt = e$ (the largest possible amount an agent pays). Since $x^* \geq \frac{E}{2} = \frac{e}{2-2\beta}$, we only need to make sure $\frac{\beta e}{2-2\beta} \geq e$. Therefore, if $\beta \geq \frac{2}{3}$, then M^* is well-defined. For specific priors, M^* may be well-defined for smaller β.

Next, we show that (whenever M^* is well-defined) M^* is competitive against the first-best allocation mechanism for *all* prior distribution f. Naturally, the competitive ratio is slightly worse than the one obtained previously for high/low valuations. We first generalize the definition of competitiveness appropriately.

Definition 7. Let W be the expected social welfare under a feasible mechanism M. Let W^F be the expected social welfare under the first-best allocation mechanism. If $W \geq \alpha W^F$ for all prior distributions, then we say that M is α-competitive. We call α a *competitive ratio* of M.

Proposition 58. *Whenever M^* is well-defined for all prior distributions (*e.g., $\beta \geq \frac{2}{3}$*), M^* is* 0.85*-competitive.*

Proof. First, we note that P (the expected payment made by an agent) equals $B/2$ (B is the expected value of the lower valuation).

$$2P = 2\int_0^\infty \int_0^v tf(t)dtf(v)dv =$$
$$\int_0^\infty \int_0^v tf(t)dtf(v)dv + \int_0^\infty \int_0^t vf(v)dvf(t)dt =$$
$$\int_0^\infty \int_0^\infty \min(t,v)f(t)f(v)dtdv.$$

The ratio between the expected social welfare under M^* and the expected social welfare under the first-best allocation mechanism equals

$$\frac{\frac{T}{2} - B + \sqrt{(B - \frac{T}{2})^2 + 2eB}}{T}$$

Since $T + B = 2e$, we have $T = ue$ for $1 \leq u \leq 2$. We may rewrite the ratio as

$$\frac{\frac{ue}{2} - (2e - ue) + \sqrt{((2e - ue) - \frac{ue}{2})^2 + 2e(2e - ue)}}{ue}$$

$$= \frac{\frac{u}{2} - (2 - u) + \sqrt{((2 - u) - \frac{u}{2})^2 + 2(2 - u)}}{u}$$

Let $t = \frac{1}{u}$ (here, $\frac{1}{2} \leq t \leq 1$). The above ratio can be rewritten in terms of t as follows:

$$\frac{3}{2} - 2t + \sqrt{(2t - \frac{3}{2})^2 + 4t^2 - 2t}$$

$$= \frac{3}{2} - 2t + \sqrt{8t^2 - 8t + \frac{9}{4}}$$

The minimal value of the above expression can occur only when t is on the boundary or the first derivative is 0.

The first derivative of the above expression is

$$-2 + \frac{8t - 4}{\sqrt{8t^2 - 8t + \frac{9}{4}}}$$

After algebraic manipulation, we find that the above first derivative equals 0 only when $32t^2 - 32t + 7 = 0$, that is, when $t = \frac{1}{2} \pm \frac{\sqrt{2}}{8}$.

Thus, we compare the values of the above expression when t equals $\frac{1}{2}$, 1, and $\frac{1}{2} + \frac{\sqrt{2}}{8}$ (we recall that t has to be at least $\frac{1}{2}$). The minimal value equals $\frac{1}{2} + \frac{\sqrt{2}}{4} \approx 0.85$ (for $t = \frac{1}{2} + \frac{\sqrt{2}}{8}$). □

3.1.5 Three or More Agents

We have focused on allocation problems with two agents. In this subsection, we generalize our analytical approach to cases of three or more agents.

Let n be the number of agents. We will continue with the state-based approach. That is, a mechanism (state) is denoted by a vector of n nonnegative real values. For example, if under mechanism M, agent i's long-term expected utility equals x_i, then mechanism M is denoted by $(x_1, x_2, \ldots, x_n)$. If we are about to apply mechanism M, then we say the agents are in state $(x_1, x_2, \ldots, x_n)$.

For any n, it is easy to see that the set of feasible states is convex and symmetric with respect to permutations of the agents. A state is called *fair* if all its elements are equal. For example, $(1,1,1)$ is a fair state ($n=3$). When there is no ambiguity about the number of agents, the fair state $(x, x, \ldots, x)$ is denoted simply by x.

An artificial payment system can be constructed in a way that is similar to the case of two agents. Let μ_{n-1} be any feasible fair state for the case of $n-1$ agents. Then, the following n states are also feasible for the case of n agents:

$$(0, \underbrace{\mu_{n-1}, \ldots, \mu_{n-1}}_{n-1}), (\mu_{n-1}, 0, \underbrace{\mu_{n-1}, \ldots, \mu_{n-1}}_{n-2}), \ldots, (\underbrace{\mu_{n-1}, \ldots, \mu_{n-1}}_{n-1}, 0).$$

We denote the above n states by s_i for $i = 1, 2, \ldots, n$. Let $\hat{S}$ be the set of all feasible states with at least one element that equals 0. $\hat{S}$ is self-generating. Suppose we have a fair state μ_n for the case of n agents. Let S be the smallest convex set containing μ_n and all the states in $\hat{S}$. The s_i are in both $\hat{S}$ and S. An artificial payment system can be implemented as follows (for the case of n agents): The agents will go to state μ_n by default. If for all i, agent i chooses to pay v_i units of artificial currency, then we move to a new state whose i^{th} element equals $\mu_n - \frac{v_i}{\beta} + \gamma \sum_{j \neq i} \frac{v_j}{\beta}$. Here $\gamma = \frac{\mu_{n-1} - \mu_n}{\mu_n}$.[7] The new state M is in S. (The reason is the following. If only agent i is paying, and it is paying nv_i instead of v_i, then the new state M_i is $(\underbrace{\mu_n + \gamma\frac{nv_i}{\beta}, \ldots, \mu_n + \gamma\frac{nv_i}{\beta}}_{i-1}, \mu_n - \frac{nv_i}{\beta}, \underbrace{\mu_n + \gamma\frac{nv_i}{\beta}, \ldots, \mu_n + \gamma\frac{nv_i}{\beta}}_{n-i})$, which is a convex

[7] It should be noted that when one agent pays 1, then *every* other agent receives γ. In a sense, γ already incorporates the fact that the payment must be divided among multiple agents.

combination of μ_n and s_i. The average of the M_i over all i is just M. Thus M is a convex combination of μ_n and the s_i, which implies $M \in S$.[8])

With the above artificial payment system, by allocating the item to the agent with the highest reported value and charging the agents dAGVA payments, we get an incentive compatible mechanism. We denote agent i's reported value by v_i for all i. The dAGVA payment for agent i equals $E_{v_{-i}}\{\max\{v_{-i}\}I(v_i \geq \max\{v_{-i}\})\}$, where I is the characteristic function (which evaluates to 1 on *true* and to 0 otherwise) and v_{-i} is the set of reported values from agents other than i.

We still use P to denote the expected amount of payment from an agent. We use T to denote the expected value of the highest reported value. The expected utility for an agent is then $\frac{T}{n} - P + (n-1)\frac{\mu_{n-1}-\mu_n}{\mu_n}P + \beta\mu_n$.

To show S is self-generating, we only need to show μ_n is in $Gen(S)$. That is, μ_n is a feasible fair state as long as μ_n satisfies the following inequality: $\mu_n \leq \frac{T}{n} - P + (n-1)\frac{\mu_{n-1}-\mu_n}{\mu_n}P + \beta\mu_n$.

The largest solution of μ_n equals $\frac{\frac{T}{n}-nP+\sqrt{(nP-\frac{T}{n})^2+4(1-\beta)(n-1)\mu_{n-1}P}}{2(1-\beta)}$.

The above expression increases when the value of μ_{n-1} increases. The highest value for μ_1 is E (when there is only one agent, we can simply give the item to the agent for free). A natural way of solving for a good fair state μ_n is to start with $\mu_1 = E$, then apply the above technique to solve for μ_2, then μ_3, *etc.*

Next, we present a proposition that is similar to Proposition 58.

Proposition 59. *Let n be the number of agents. Let M_n^* be the mechanism obtained by the technique proposed in this subsection. Whenever $\beta \geq \frac{n^2}{n^2+\frac{3}{4}}$, M_n^* is well defined for all priors, and is α_n-competitive, where $\alpha_1 = 1$, and for $n > 1$,*

$$\alpha_n = \min_{\{1 \leq u \leq \frac{n}{n-1}\}} n\frac{\frac{u}{n}-n+nu-u+\sqrt{(n-nu+u-\frac{u}{n})^2+4\alpha_{n-1}\frac{n-nu+u}{n}}}{2u}.$$

[8] The above argument assumes that the available budget is at least n times the maximum amount an agent pays.

For all i, $\alpha_i \geq \frac{3}{4}$ holds.

Proof. It is straightforward that $\alpha_1 = 1$.

Given the value of α_{n-1}, we have that $(n-1)\mu_{n-1} \geq \alpha_{n-1}\frac{T_{n-1}}{1-\beta}$, where T_{n-1} is the expected value of the largest valuation among $n-1$ agents. We know that larger μ_{n-1} result in larger μ_n. That is, μ_n is at least

$$\frac{\frac{T_n}{n} - nP + \sqrt{(nP - \frac{T_n}{n})^2 + 4(1-\beta)\alpha_{n-1}\frac{T_{n-1}}{1-\beta}P}}{2(1-\beta)}$$

Here, T_n is the expected value of the largest valuation among n agents. P is the expected value of the dAGVA payment for the case of n agents.

The ratio between the expected social welfare under M_n^* and the expected social welfare under the first-best allocation mechanism is then at least

$$n\frac{\frac{T_n}{n} - nP + \sqrt{(nP - \frac{T_n}{n})^2 + 4\alpha_{n-1}T_{n-1}P}}{2T_n}$$

Through algebraic manipulation, we have that $nT_{n-1} = (n-1)T_n + nP$. That is, we may rewrite P in terms of T_{n-1} and T_n. Because of this, on the one hand, we have $T_n \geq T_{n-1}$. On the other hand, we have $T_n \leq \frac{n}{n-1}T_{n-1}$. We rewrite T_n as uT_{n-1} for $1 \leq u \leq \frac{n}{n-1}$. After simplification, the above ratio becomes

$$n\frac{\frac{u}{n} - n + nu - u + \sqrt{(n - nu + u - \frac{u}{n})^2 + 4\alpha_{n-1}\frac{n-nu+u}{n}}}{2u}$$

The competitive ratio of M_n^* is at least the minimal value of the above expression.

We next prove that $\alpha_i \geq \frac{3}{4}$ for all i, by induction. We already have $\alpha_1 = 1 \geq \frac{3}{4}$. Now we prove that for any i, if $\alpha_{i-1} \geq \frac{3}{4}$, then we also have $\alpha_i \geq \frac{3}{4}$.

For α_i with $i \geq 2$, if $\alpha_{i-1} \geq \frac{3}{4}$, we have

$$\alpha_i = \min_{\{1 \leq u \leq \frac{i}{i-1}\}} i\frac{\frac{u}{i} - i + iu - u + \sqrt{(i - iu + u - \frac{u}{i})^2 + 4\alpha_{i-1}\frac{i-iu+u}{i}}}{2u}$$

$$\geq \min_{\{1\leq u\leq \frac{i}{i-1}\}} i\frac{\frac{u}{i}-i+iu-u+\sqrt{(i-iu+u-\frac{u}{i})^2+3\frac{i-iu+u}{i}}}{2u}$$

$$= \min_{\{1\leq u\leq \frac{i}{i-1}\}} (\frac{i^2-i+1}{2}-\frac{i^2}{2u}+\sqrt{(\frac{i^2}{2u}-\frac{i^2}{2}+\frac{i}{2}-\frac{1}{2})^2+3(\frac{i^2}{4u^2}-\frac{i^2}{4u}+\frac{i}{4u})})$$

Let $t=\frac{i}{u}-i+1$ $(0\leq t\leq 1)$. We have

$$\alpha_i \geq \min_{\{0\leq t\leq 1\}} (\frac{1}{2}-\frac{i}{2}t+\sqrt{(\frac{i}{2}t-\frac{1}{2})^2+\frac{3}{4}t(t+i-1)})$$

$$= \min_{\{0\leq t\leq 1\}} (\frac{1}{2}-\frac{i}{2}t+\sqrt{(\frac{i}{2}t-\frac{1}{2})^2+\frac{3it}{4}+\frac{3}{4}(t^2-t)})$$

$$\geq \min_{\{0\leq t\leq 1\}} (\frac{1}{2}-\frac{i}{2}t+\sqrt{(\frac{i}{2}t-\frac{1}{2})^2+\frac{3it}{4}+\frac{3}{4}(-\frac{1}{4})})$$

$$= \min_{\{0\leq t\leq 1\}} (\frac{1}{2}-\frac{i}{2}t+\sqrt{\frac{i^2t^2}{4}-\frac{it}{2}+\frac{1}{4}+\frac{3it}{4}-\frac{3}{16}})$$

$$= \min_{\{0\leq t\leq 1\}} (\frac{1}{2}-\frac{i}{2}t+\sqrt{\frac{i^2t^2}{4}+\frac{it}{4}+\frac{1}{16}})=\frac{3}{4}$$

Therefore, by induction, we have proved that $\alpha_i \geq \frac{3}{4}$ for all i.

When there are n agents, for M_n^* to be well-defined, we need M_i^* to be well-defined for all $i \leq n$, because M_{i+1}^* is constructed based on M_i^*. That is, we need the budget $\beta\mu_i$ to be greater than or equal to iT_{i-1} for all $i \leq n$. Here, T_{i-1} is the largest possible amount an agent pays when there are i agents, and the budget must be at least i times this amount (Footnote 8). Since $\mu_i \geq \frac{(i-1)\mu_{i-1}}{i} \geq \frac{\alpha_{i-1}T_{i-1}}{i-i\beta}$, we need to make sure $\frac{\beta\alpha_{i-1}}{i-i\beta} \geq i$. This is satisfied if $\beta \geq \frac{i^2}{i^2+\alpha_{i-1}}$. That is, as long as $\beta \geq \max_{\{i\leq n\}}\{\frac{i^2}{i^2+\alpha_{i-1}}\}$, M_n^* is well-defined. Since $\alpha_i \geq \frac{3}{4}$ for all i, we have that as long as $\beta \geq \frac{n^2}{n^2+\frac{3}{4}}$, M_n^* is well-defined. (For specific priors, M_n^* may be well-defined for smaller β.) □

As a comparison, the lottery mechanism that always chooses the winner uniformly at random has competitive ratio (exactly) $\frac{1}{n}$, which goes to 0 as n goes to infinity.

3.2 Competitive Multi-Item Allocation Without Payments

In the previous section, we studied repeated allocation without payments. In this section, we study single-round allocation without payments. In this context, if we try to allocate a single item, there is little that can be done, because there is nothing that an agent can give up in order to get the item. Therefore, in this section, we study allocating multiple heterogeneous items. We investigate the problem of allocating items (private goods) among competing agents in a setting that is both prior-free and payment-free. That is, we do not assume that we have knowledge about the distribution of the agents' valuations. We also do not allow the mechanism to specify any monetary payments. Specifically, we focus on allocating multiple heterogeneous items between two agents. Our objective is to design strategy-proof mechanisms that are competitive against the efficient (first-best) allocation.

It remains an open question to give an elegant characterization of mechanisms that are strategy-proof, prior-free, and payment-free (for the problem that we study), and we do not know how to solve for the most competitive such mechanism in general. In our attempts to design competitive mechanisms, we introduce the family of *linear increasing-price (LIP) mechanisms*, which are based on a certain artificial currency. The LIP mechanisms are strategy-proof, prior-free, and payment-free. We show how to solve for competitive mechanisms within the LIP family. For the case of two items, we find a LIP mechanism whose competitive ratio is near optimal (the achieved competitive ratio is 0.828, while any strategy-proof mechanism is at most 0.841-competitive). Thus, at least for the case of two items, it does not come at much of a loss to focus only on LIP mechanisms. As the number of items goes to infinity, we prove a negative result that any increasing-price mechanism (linear or nonlinear) has a maximal competitive ratio of 0.5.

This section is organized as follows. In Subsection 3.2.1, we describe the model that we study. In Subsection 3.2.2, we prove an upper bound on the competitive ratios of strategy-proof mechanisms. In Subsection 3.2.3, we introduce the family of linear increasing-price mechanisms, and show they are the only increasing-price mechanisms satisfying a strong responsiveness property. In Subsection 3.2.4, we solve for competitive mechanisms within the LIP family. Finally, in Subsection 3.2.5, we prove a negative result that as the number of items goes to infinity, any increasing-price mechanism (linear or nonlinear) has a maximal competitive ratio of 0.5.

3.2.1 Formalization

We study the problem of allocating m ($m > 1$) heterogeneous items (referred to as items 1 to m) between two agents (referred to as agents 1 and 2). We use $-i$ to denote the agent other than i.[9]

We still use O to denote the set of all possible allocations (outcomes). For the problem that we study here, a specific allocation $o \in O$ is denoted by a vector $(p_1, p_2, \ldots, p_m)$ ($0 \le p_j \le 1$ for all j), where p_j is the proportion[10] of item j won by agent 1 (so that $1 - p_j$ is the proportion of item j won by agent 2).

We assume that the agents' valuations for the items are additive. We use a vector $(v_1^i, v_2^i, \ldots, v_m^i)$ to denote agent i's type, where v_j^i is agent i's valuation for winning item j ($v_j^i \ge 0$). Additivity implies that under allocation $(p_1, p_2, \ldots, p_m)$, agent 1's utility equals $\sum_j p_j v_j^1$ and agent 2's utility equals $\sum_j (1 - p_j) v_j^2$.

Furthermore, we require that the agents' valuations are normalized. That is, the type space Ω consists of vectors $(v_1, v_2, \ldots, v_m)$ with $\sum_j v_j = 1$. As a result, an agent's utility for an allocation can be thought of as her level of satisfaction; if an agent wins all the items, then she is 100% satisfied. The reason that we require this normalization is the following. When payments are available and utility

[9] We usually use $-i$ to denote the set of all agents other than i. Here, since there are only two agents, we simply use $-i$ to denote the agent other than i.

[10] For indivisible items, p_j is interpreted as the probability that agent 1 wins item j.

is quasi-linear, payments provide a way of comparing valuations between agents. However, because payments are unavailable in our context, it is no longer possible to make such a comparison. Hence, the units in which valuations are expressed become meaningless, so that the only meaning that can be derived from an agent's valuations is the *relative* valuations of the items (the ratio of the valuations). If we (say) doubled one agent's valuation for every item, in our payment-free context this would double that agent's utility for every outcome, and as a result her behavior under any mechanism would remain completely unchanged. As a result, there can be no hope of coming anywhere close to maximizing the social welfare without some normalization assumption.

We define the *first-best allocation mechanism* M^* to be the mechanism that always naïvely maximizes the social welfare (without considering incentives). We will use the first-best mechanism M^* (which is not strategy-proof) as our benchmark when evaluating the performance of strategy-proof mechanisms. (When using M^* as a benchmark, we assume that agents report truthfully, even though they are not incentivized to do so. Hence, M^* always produces the maximal social welfare among all mechanisms, with or without priors, and with or without payments.)

Strategy-proof mechanism M is said to be (at least) α*-competitive* if the social welfare under M is always greater than or equal to α times the social welfare under M^*. Here α is called M's *competitive ratio*. The maximal possible value of α is called M's *maximal competitive ratio*.

Example 15. The mechanism that always divides every item evenly has maximal competitive ratio 0.5. The mechanism that always gives every item to agent 1 also has maximal competitive ratio 0.5.

Our objective is to design strategy-proof (payment-free) mechanisms with high competitive ratios.

3.2.2 Upper Bound on the Competitive Ratios

In this subsection, we derive an upper bound on the competitive ratios of strategy-proof mechanisms. Given our objective, we only need to consider strategy-proof mechanisms that are *symmetric*.[11]

Definition 8. A mechanism M is symmetric if it satisfies

Symmetry over the agents: If we swap the reported type vectors of two of the agents, then the items allocated to these agents are also swapped.

Symmetry over the items: If we swap agent 1's valuations for any two items, and we swap agent 2's valuations for the same two items, then the allocation result for these two items is also swapped.

Proposition 60. *For any strategy-proof mechanism that is α-competitive, there is a corresponding* symmetric *strategy-proof mechanism that is (at least) α-competitive.*

Proof. Let M be any strategy-proof mechanism. Let $M_{\sigma_2}^{\sigma_1}$ be the mechanism that is the same as M, except that the indices of the agents are permuted according to an arbitrary permutation σ_1, and the indices of the items are permuted according to an arbitrary permutation σ_2. Let $\bar{M}$ be the "average" of the mechanisms $M_{\sigma_2}^{\sigma_1}$ over all permutations σ_1 and σ_2. (That is, with equal probability, $\bar{M}$ is the mechanism corresponding to a specific permutation of the agents and the items.) $\bar{M}$ is strategy-proof, symmetric, and at least α-competitive. □

Proposition 61. *For the case of two agents, any symmetric strategy-proof mechanism is (at least)* 0.5*-competitive.*

Proof. By symmetry over the agents, given an agent's report, the other agent can guarantee herself a utility of 0.5, by cloning the report of the first agent. Because

[11] This is a frequently used technique in the literature on prior-free mechanism design.

the mechanism is strategy-proof, it follows that if the agents report truthfully, then each of them has utility at least 0.5. Hence, the social welfare under any symmetric strategy-proof mechanism is at least 1. The social welfare under the first-best mechanism is at most 2. □

Proposition 60 implies that for the purpose of deriving an upper bound on the competitive ratios of strategy-proof mechanisms, we can safely ignore strategy-proof mechanisms that are not symmetric.

Let us recall that a mechanism M is α-competitive if for *all possible* type vectors, the social welfare under M is at least α times the social welfare under the first-best mechanism M^*. If we restrict the type space, then the maximal competitive ratio of M can only stay the same or increase. That is, one way to compute an upper bound on the competitive ratios of strategy-proof mechanisms is to restrict the type space and then solve for the largest possible competitive ratio for any strategy-proof mechanism.

Theorem 10. *The competitive ratio of any strategy-proof mechanism is at most* 0.841*. This is true for any number of items and two agents.*

Proof. We first focus on the case of two items. We consider the following restricted type space: $\{(ih, (N-i)h) | i = 0, 1, \ldots, N\}$, where $N = 50$ and $h = 1/N$. Type vector $(ih, (N-i)h)$ can be denoted by the integer i. A mechanism for this restricted type space can be denoted by the p^i_{jk} for $i = 1, 2$ and $0 \leq j, k \leq N$, where p^i_{jk} is the proportion of item i won by agent 1 when agent 1's report is j and agent 2's report is k.

Strategy-proofness for agent 1 can then be represented by the following set of linear inequalities: $\forall\ 0 \leq j, j', k \leq N$

$$jp^1_{jk} + (N-j)p^2_{jk} \geq jp^1_{j'k} + (N-j)p^2_{j'k}$$

Strategy-proofness for agent 2 can be represented by a similar set of linear inequalities involving the p^i_{jk}.

The mechanism characterized by the p^i_{jk} is α-competitive if the following linear inequalities are satisfied: $\forall\ 0 \leq j, k \leq N$

$$jp^1_{jk} + (N-j)p^2_{jk} + k(1-p^1_{jk}) + (N-k)(1-p^2_{jk}) \geq$$

$$\alpha(\max\{j,k\} + \max\{N-j, N-k\})$$

The largest possible competitive ratio for any mechanism and for the above restricted type space can thus be computed by solving a linear program, which results in 0.841.[12] Any strategy-proof mechanism for the case of $m > 2$ items remains strategy-proof when applied to the case of two items (when the agents do not care about the other items). Hence, the upper bound 0.841 still applies. □

3.2.3 Linear Increasing-Price Mechanisms

As mentioned earlier, it remains an open question to solve for the most competitive strategy-proof mechanism in general. There are two reasons for this: first, we lack an elegant characterization of all strategy-proof mechanisms for our problem; second, we lack a general approach for evaluating a given mechanism (computing its maximal competitive ratio).

In our attempts to design competitive mechanisms, we start with the family of all strategy-proof mechanisms (SP). We then move on to more and more restricted families of mechanisms: the family of swap-dictatorial mechanisms (SD), the family of increasing-price mechanisms (IP), and finally the family of linear increasing-price

[12] We acknowledge that a computer-assisted proof is not as satisfactory as an easily human-verifiable mathematical proof. Because this is a linear programming problem, in principle, we can give a (nearly) optimal solution to the dual problem to show that it is impossible to better; we do not give such a solution here because it does not seem to shed much light.

mechanisms (LIP). These 4 families are nested as illustrated below:

$$LIP \subsetneq IP \subsetneq SD \subsetneq SP$$ [13]

As we move from SP to LIP, we get more and more elegant characterizations of the mechanisms. Finally, the mechanisms in the LIP family can actually be characterized by a single parameter, and we are able to evaluate (the competitiveness of) any given LIP mechanism. That is, we are able to solve for competitive mechanisms within the LIP family.

In a payment-free setting, if we fix agent $-i$'s report, then agent i essentially faces a set of allowable outcomes that she can choose from (each outcome corresponds to an allowable report of i). A necessary condition for a mechanism to be strategy-proof is that the mechanism should always choose i's favorite outcome (among all allowable outcomes). This condition is not sufficient for the mechanism to be strategy-proof for *both* agents, because agent $-i$ may have the power to change the set of allowable outcomes that agent i faces. That is, $-i$ may want to submit a false report to get agent i to a decision $-i$ prefers. However, if we require that the set of allowable outcomes agent i faces is fixed, then the mechanism that picks i's favorite outcome is strategy-proof for both agents. Essentially, in such a mechanism, agent i is the dictator: she chooses her favorite outcome from a set of outcomes predetermined by the mechanism, and agent $-i$ has no choice but to accept this outcome (the decision is solely made by i). This leads to the following family of swap-dictatorial mechanisms (by Proposition 60, we only need to consider symmetric mechanisms):

Swap-Dictatorial Mechanisms: With probability 0.5, agent i is the dictator, who chooses her favorite allocation from a predefined set of allowable allocations $\hat{O}^i \subset O$. The $\hat{O}^i$ satisfy the following (symmetry over the agents and the items):

[13] If we define SP to be the family of all *symmetric* strategy-proof mechanisms, then we can only say SD is a subset of SP. We do not know whether it is a strict subset or not.

- If $(p_1, p_2, \ldots, p_m) \in \hat{O^i}$, then $(1 - p_1, 1 - p_2, \ldots, 1 - p_m) \in \hat{O^{-i}}$ for any i.
- If $(p_1, p_2, \ldots, p_m) \in \hat{O^i}$, then $(p_{\sigma(1)}, p_{\sigma(2)}, \ldots, p_{\sigma(m)}) \in \hat{O^i}$ for any permutation σ and i.

Swap-dictatorial mechanisms, as well as other dictatorial mechanisms, have been studied extensively because of their simplicity (*e.g.*, [17]). Many papers in the literature on mechanisms without payments suggest that strategy-proofness, combined with various other properties, can only come down to mechanisms that are dictatorial in nature [89, 44, 112]. However, since we do not assume additional properties, for our problem, there do exist strategy-proof mechanisms that are not dictatorial in nature as shown below:

Example 16. Example fixed price mechanism [10] (strategy-proof, but not dictatorial): There are two items. Let $r \in [1, \infty)$ be a constant. The mechanism starts by assigning both agents half of each item. Agent 1 has the right to propose the following trade: sell $\frac{1}{2}$ units of item 1 to agent 2 for $\frac{1}{2r}$ units of item 2. (Agent 1 does not have to propose the trade.) Agent 2 then approves or vetoes the trade proposal.

First of all, it can be seen that this is not a deterministic dictatorial mechanism. Under a deterministic dictatorial mechanism, the dictator and the set of outcomes the dictator can choose from are both predetermined precisely. That is, under a deterministic dictatorial mechanism, at most one agent can affect the allocation (it could be that the dictator agent faces one choice only). On the other hand, under the above fixed price mechanism, the allocation depends on *both* agents' reports. Next, we show that the above fixed price mechanism is not equivalent to a randomization over deterministic dictatorial mechanisms (two mechanisms are equivalent if they always produce the same allocation). We assume that the above fixed price mechanism is equivalent to a randomization over deterministic dictatorial mechanism $M_1, M_2, \ldots, M_x$. Without loss of generality, we assume under $M_1, M_2, \ldots, M_k$

($k \leq x$), agent 1 is the dictator, and under $M_{k+1}, M_{k+2}, \ldots, M_x$, agent 2 is the dictator. Essentially, we are assuming that the above fixed price mechanism is equivalent to a randomization over two mechanisms M^1 and M^2: under M^i, agent i is the dictator who is choosing from a randomized set of outcomes. When agent 1's type is $(1, 0)$, that is, when agent 1 is only interested in item 1, she will not propose the trade under the fixed price mechanism. That is, the final allocation is fixed to $(0.5, 0.5)$. When agent 1's type is fixed to $(1, 0)$, the allocation under M^1 is fixed. Therefore, when agent 1's type is fixed to $(1, 0)$, the allocation under M^2 is also fixed regardless of agent 2's type. However, the allocation under M^2 does not depend on agent 1's type. That is, the allocation under M^2 is fixed regardless of both agents' types. Similarly, given our assumption, we can see that the allocation under M^1 is also fixed regardless of both agents' types. That is, the above fixed price mechanism must produce the same allocation regardless of both agents' types. However, this is clearly not the case. By contradiction, the above fixed price mechanism is not equivalent to a randomization over deterministic dictatorial mechanisms. In conclusion, we proved that not all strategy-proof mechanisms are randomizations over dictatorial mechanisms. However, we do not know whether all *symmetric* strategy-proof mechanisms are randomizations over dictatorial mechanisms or not.

For purpose of maximizing social welfare, ideally, we want the dictator agent to take only items that she really values, and leave the remaining items to the other agent. This leads to the following family of *increasing-price (IP) mechanisms.*

Increasing-Price (IP) Mechanisms: With probability 0.5, agent i is the dictator, and is endowed with 1 unit of artificial currency. The dictator agent can purchase (proportions of) items (from the mechanism, not from the other agent) with her artificial currency. The (proportions of) items not purchased at the end go to the other agent. Rather than having just a fixed price for each item, there is a price schedule for each item, and the item becomes more expensive as the dictator agent buys more of it. The price schedules are characterized by functions $f_j^i : [0,1] \to \mathbb{R}^+$ for $i = 1, 2$ and $j = 1, 2, \ldots, m$. $f_j^i(x)$ is the instantaneous price per unit charged to agent i (when i is the dictator) if she demands item j, at the point where x units of her artificial currency have already been spent on item j. By Proposition 60, we can simply assume $f_j^i = f$ for all i and j. Function f is increasing and positive. We also assume f is differentiable. If, at the end, agent i (when she is the dictator) spent x units of artificial currency on item j, then she is allocated a proportion $\int_0^x \frac{1}{f(t)}dt$ of item j. We will present an example IP mechanism later in this subsection (which actually belongs to the more restricted class of LIP mechanisms).

The intuition for why increasing-price mechanisms might perform well is as follows. If the dictator agent demands a large proportion of an item, then she will be paying at a high rate, which can only happen when she highly values the item. Because prices are increasing, the optimal strategy for the dictator agent is simply the greedy strategy: purchase (an infinitesimally small amount each time) the best deal (the item with the highest value/price ratio) until the artificial currency runs out. That is, at some point, if the dictator agent's valuation for item j is v_j, and so far x_j units of artificial currency have been spent on item j, then the dictator agent should purchase an infinitesimally small amount of item j^*, where $j^* = \arg\max_j \{\frac{v_j}{f(x_j)}\}$. At the end, for items that have been partly purchased, the final prices must be proportional to the dictator agent's valuations:

Lemma 6. *Under an IP mechanisms, if the dictator spends $k_1, k_2 (0 < k_i < 1)$ units of artificial currency on items $1, 2$, then the dictator's valuations for these items must be $f_1(k_1) \cdot C$ and $f_2(k_2) \cdot C$ for some C.*

Any increasing and positive function f corresponds to an increasing-price mechanism. Actually, for the purpose of designing competitive mechanisms, we only need to consider functions f that satisfy $\int_0^1 \frac{1}{f(t)}dt = 1$. That is, we only need to consider increasing-price mechanisms in which the dictator agent gets the entirety of an item if and only if she spends all her artificial currency on this item.

Proposition 62. *For the purpose of designing competitive IP mechanisms, we only need to consider increasing-price mechanisms with f satisfying $\int_0^1 \frac{1}{f(t)}dt = 1$.*

Proof. If $\int_0^1 \frac{1}{f(t)}dt > 1$, then there exists U ($U < 1$) that satisfies $\int_0^U \frac{1}{f(t)}dt = 1$. $\forall 0 < \epsilon < U$, let $\hat{f}$ be the same as f for $x \leq U$, and let $\hat{f}(x)$ take some very high values for $U < x \leq 1$ (in a way that makes $\hat{f}$ increasing), so that $\int_0^1 \frac{1}{\hat{f}(t)}dt \leq 1 + \epsilon$. Since the dictator agent will never spend more than U units of artificial currency on any item (it is pointless for the dictator agent to continue purchasing an item when she has already obtained the entirety of this item), on the region that matters to the mechanism ($0 \leq x \leq U$), f and $\hat{f}$ are identical. Thus, we only need to consider functions f satisfying $\int_0^1 \frac{1}{f(t)}dt \leq 1 + \epsilon$ for arbitrary small value ϵ. That is, we only need to consider cases where $\int_0^1 \frac{1}{f(t)} \leq 1$.

If $\int_0^1 \frac{1}{f(t)}dt = p < 1$, then let $\hat{f} = pf$, so that we have $\int_0^1 \frac{1}{\hat{f}(t)}dt = 1$. We denote the proportion of item j won by agent i under f when i is the dictator by q_j^i. The proportion of item j won by agent i under f when i is not the dictator is then $1 - q_j^{-i}$. The proportion of item j won by agent i under $\hat{f}$ when i is the dictator is $\frac{q_j^i}{p}$ (under $\hat{f}$, a dictator gets $\frac{1}{p}$ times as much item per unit of artificial currency at every amount of currency spent), and the proportion of item j won by agent i under $\hat{f}$ when i is not the dictator is $1 - \frac{q_j^{-i}}{p}$. The social welfare under f equals $\sum_{i,j} \frac{q_j^i v_j^i + (1-q_j^{-i})v_j^i}{2}$. The social welfare under $\hat{f}$ equals $\sum_{i,j} \frac{q_j^i v_j^i/p + (1-q_j^{-i}/p)v_j^i}{2}$, which is at least 1 (as in the proof of Proposition 61). It turns out that the social welfare under f is always less than or equal to the social welfare under $\hat{f}$, as proved below. $\sum_{i,j} \frac{q_j^i v_j^i + (1-q_j^{-i})v_j^i}{2} = \sum_{i,j} \frac{q_j^i v_j^i + (p-q_j^{-i})v_j^i + (1-p)v_j^i}{2} = \sum_{i,j} \frac{q_j^i v_j^i + (p-q_j^{-i})v_j^i}{2} + \sum_{i,j} \frac{(1-p)v_j^i}{2} =$

$\sum_{i,j} \frac{q_j^i v_j^i + (p - q_j^{-i}) v_j^i}{2} + (1-p) = p \sum_{i,j} \frac{q_j^i v_j^i / p + (1 - q_j^{-i}/p) v_j^i}{2} + (1-p) \leq \sum_{i,j} \frac{q_j^i v_j^i / p + (1 - q_j^{-i}/p) v_j^i}{2}$.

Hence, we only need to consider f satisfying $\int_0^1 \frac{1}{f(t)} = 1$. □

Finally, the family of linear increasing-price mechanisms is described below:

Linear Increasing-Price (LIP) Mechanisms: Linear increasing-price mechanisms are increasing-price mechanisms characterized by a linear function $f(x) = ax + b$, where a and b are positive constants. (a has to be positive for f to be increasing. b has to be positive to avoid negative prices or division-by-zero.) Since we only consider f satisfying $\int_0^1 \frac{1}{f(t)} dt = 1$, we have $b = \frac{a}{e^a - 1}$. That is, a LIP mechanism is characterized by a single parameter a. From now on, we use $LIP(a)$ to denote the LIP mechanism with parameter a. We use b to denote the value $\frac{a}{e^a-1}$.

Example 17. Let $a = 2$ ($b = \frac{2}{e^2-1}$) and $m = 2$. Let the agents' type vectors be $(1, 0)$ and $(0.5, 0.5)$, respectively. Under $LIP(a)$, with 0.5 probability, agent 1 is the dictator. Since agent 1's type vector is $(1, 0)$, she will spend all her artificial currency on item 1. The resulting allocation is $(1, 0)$: agent 1 wins the entirety of item 1, while agent 2 gets what is left (the entirety of item 2). With 0.5 probability, agent 2 is the dictator. Since agent 2's type vector is $(0.5, 0.5)$, she will divide her artificial currency evenly on items 1 and 2. The resulting allocation is $(1 - \int_0^{0.5} \frac{1}{at+b} dt, 1 - \int_0^{0.5} \frac{1}{at+b} dt) = (0.283, 0.283)$: agent 2 wins $\int_0^{0.5} \frac{1}{at+b} dt = 0.717$ proportion of both item 1 and 2, while agent 1 gets what is left ($1 - \int_0^{0.5} \frac{1}{at+b} dt = 0.283$ proportion of both items). In total, the resulting allocation under $LIP(a)$ is $(1 - \frac{1}{2} \int_0^{0.5} \frac{1}{at+b} dt, \frac{1}{2} - \frac{1}{2} \int_0^{0.5} \frac{1}{at+b} dt) = (0.642, 0.642)$.

Besides simplicity, the linear increasing-price mechanisms possess a nice property that is not shared by other (non-linear) increasing-price mechanisms. Before defining this property, we need the following definitions. Suppose we are considering an IP mechanism characterized by function f.

Definition 9. A type vector $\vec{v} \in \Omega$ is *strictly full ranked* for f if a dictator agent with true type $\vec{v}$ will purchase positive proportions of every item under f.

Every strictly full ranked type vector $\vec{v} = (v_1, v_2, \ldots, v_m)$ corresponds to a vector $(t_1, t_2, \ldots, t_m)$ with $\sum_{j=1}^{m} t_j = 1$, where t_j (> 0) denotes the amount of artificial currency that an agent with type vector $\vec{v}$ will spend on item j (when she is the dictator). The final value/price ratio $\frac{v_j}{f(t_j)}$ should be the same for all j (Lemma 6).

Definition 10. A type vector $\vec{v} \in \Omega$ is *full ranked* if $\vec{v} \in \mathbb{W}$, where $\mathbb{W}$ is the closure of the set of all strictly full ranked type vectors.

For a full ranked vector $\vec{v}$, we also have that the final value/price ratio $\frac{v_j}{f(t_j)}$ should be the same for all j.

Not all type vectors are full ranked type vectors. If an agent has very low valuations for some items, then she will not spend any artificial currency on those items if $f(0)$ is sufficiently high. For small $f(0)$, most type vectors are full ranked. In the rest of this subsection (when solving for the competitive ratios of LIP mechanisms), we focus on full ranked type vectors, and treat vectors that are not full ranked as exceptions.

Proposition 63. *For cases of at least three items, LIP mechanisms are the only IP mechanisms satisfying the following condition:*

Strong responsiveness: *For two agents with full ranked type vectors, if one agent values an item more than the other agent, then she should win a greater proportion of this item than the other agent.*

We first prove the following lemma, which will be useful later.

Lemma 7. *Let $\vec{v} = (v_1, v_2, \ldots, v_m)$ be a full ranked vector under $LIP(a)$. Let $\vec{v}$'s payment vector $(t_1, t_2, \ldots, t_m)$ ($\sum_{j=1}^{m} t_j = 1$) be such that an agent with true type $\vec{v}$*

will spend t_j units of artificial currency on item j under $LIP(a)$ (when she is the dictator). Then, the v_j and the t_j satisfy $v_j = \frac{at_j+b}{a+mb}$ for all j.

Proof. The final value/price ratio $\frac{v_j}{at_j+b}$ should be the same for all j, by Lemma 6. Since $\sum v_j = 1$, we have $v_j = \frac{at_j+b}{a+mb}$ for all j. □

Now we are ready to prove the above proposition.

Proof of Proposition 63. We first prove that LIP mechanisms satisfy the strong responsiveness condition.

Lemma 7 says that under a LIP mechanism, an agent's value for an item is linear in the amount of artificial currency this agent would spend on the item as a dictator. Therefore, if one agent values an item more than the other agent, then, as the dictator, she would spend more on this item than the other agent, which means she wins more of the item at the end.

We now prove that LIP mechanisms are the only IP mechanisms satisfying the strong responsiveness condition, for cases of at least three items.

Let us consider an IP mechanism characterized by an increasing positive function f. If there exist nonnegative t_a, t_b, t'_a, t'_b, so that $0 \leq t_a + t_b = t'_a + t'_b = t \leq 1$ and $f(t_a) + f(t_b) > f(t'_a) + f(t'_b)$ are both satisfied, then we can construct the following full ranked type vectors:

$(\frac{f(1-t)}{f(1-t)+f(t_a)+f(t_b)+(m-3)f(0)}, \frac{f(t_a)}{f(1-t)+f(t_a)+f(t_b)+(m-3)f(0)}, \frac{f(t_b)}{f(1-t)+f(t_a)+f(t_b)+(m-3)f(0)}, \frac{f(0)}{f(1-t)+f(t_a)+f(t_b)+(m-3)f(0)}, \ldots, \frac{f(0)}{f(1-t)+f(t_a)+f(t_b)+(m-3)f(0)})$, and

$(\frac{f(1-t)}{f(1-t)+f(t'_a)+f(t'_b)+(m-3)f(0)}, \frac{f(t'_a)}{f(1-t)+f(t'_a)+f(t'_b)+(m-3)f(0)}, \frac{f(t'_b)}{f(1-t)+f(t'_a)+f(t'_b)+(m-3)f(0)}, \frac{f(0)}{f(1-t)+f(t'_a)+f(t'_b)+(m-3)f(0)}, \ldots, \frac{f(0)}{f(1-t)+f(t'_a)+f(t'_b)+(m-3)f(0)})$.

The two vectors are constructed in such a way that agent 1 will spend $1-t$ units of artificial currency on item 1, t_a units on item 2, t_b units on item 3, and 0 units on the other items, while agent 2 will spend $1-t$ units of artificial currency on item 1, t'_a units on item 2, t'_b units on item 3, and 0 units on the other items. Agent 1 values item 1 less than agent 2 (the denominator is larger), but they will spend the same amount of artificial currency on item 1. So, they win the same proportion of item 1 at the end. Now if we increase the value of agent 1 for item 1 by a tiny amount (still keeping it less than the value of agent 2), then we have a situation where agent 1 values item 1 less, but wins a greater proportion of it at the end (agent 1 now spends more on item 1). That is, to satisfy the strong responsiveness condition, whenever $0 \leq t_a + t_b = t'_a + t'_b = t \leq 1$ for nonnegative t_a, t_b, t'_a, t'_b, we must have $f(t_a) + f(t_b) = f(t'_a) + f(t'_b)$. That is, $\forall 0 \leq c \leq t \leq 1$, we have $f(t) + f(0) = f(t-c) + f(c)$. Since we assume f is differentiable, by taking the derivative over t on both sides of the equality, we have that $f'(t) = f'(t-c)$. The values of t and c can be arbitrary. That is, f' is a constant. f must be linear. □

The above proposition provides another justification (other than simplicity) why, among all IP mechanisms, we focus on LIP mechanisms. In what follows, we solve for competitive mechanisms within the LIP family.

3.2.4 Competitive Linear Increasing-Price Mechanisms

Since a linear increasing-price mechanism is characterized by a single parameter, if, for a given value of a, we are able to evaluate the competitiveness of $LIP(a)$, then the task of solving for competitive LIP mechanisms can be done simply by searching for the optimal value of a.

In what follows, we discuss how to evaluate the competitiveness of $LIP(a)$, for a given value of a and a given number of items.

Two Items

We first focus on the case of two items.

We denote the type vectors of agent 1 and 2 by $(x, 1-x)$ and $(y, 1-y)$, respectively

$(1 \geq x \geq y \geq 0)$. We abuse notation by using x to refer to both the value x and the type vector whose first element is x. We do the same for y.

Proposition 64. *Under $LIP(a)$, with probability 0.5, agent 1 is the dictator, whose optimal strategy (when she is the dictator) is as follows.*

- *If $\frac{x}{a+b} \geq \frac{1-x}{b}$, then agent 1 will spend all her artificial currency on item 1. At the end, agent 1 gets item 1 in its entirety while agent 2 gets what 1 does not take (item 2 in its entirety). It should be noted that this is the resulting allocation* when agent 1 is the dictator*. When agent 2 is the dictator, we may get a different allocation.*

- *If $\frac{1-x}{a+b} \geq \frac{x}{b}$, then agent 1 will spend all her artificial currency on item 2. At the end, agent 1 gets item 2 in its entirety while agent 2 gets item 1 in its entirety.*

- *Otherwise, agent 1 will spend $t = \frac{x(a+2b)-b}{a}$ units of artificial currency on item 1, and $1 - t = \frac{(1-x)(a+2b)-b}{a}$ units of artificial currency on item 2. At the end, the instantaneous prices of items 1 and 2 will be $at + b = x(a + 2b)$ and $a(1 - t) + b = (1 - x)(a + 2b)$, respectively. (We note that the prices are proportional to agent 1's type vector $(x, 1 - x)$, as they should be.) At the end, agent 1 gets a proportion $\frac{\ln(at+b)}{a} - \frac{\ln(b)}{a}$ of item 1 and a proportion $\frac{\ln(a(1-t)+b)}{a} - \frac{\ln(b)}{a}$ of item 2, while agent 2 gets the remainder.*

For $j = 1, 2$, we use $p_j(x, y)$ to denote the proportion of item j won by agent 1 at the end, when agent 1's reported type vector is x and agent 2's reported type vector is y. (This proportion takes the randomization over who is the dictator into account.) The value of $p_j(x, y)$ can be computed as shown above. $p_1(x, y)$ is increasing in x and decreasing in y. $p_2(x, y)$ is decreasing in x and increasing in y. We use $S(x, y)$ to denote the social welfare under $LIP(a)$. That is, $S(x, y) = xp_1(x, y) + (1 -$

$x)p_2(x,y) + y(1-p_1(x,y)) + (1-y)(1-p_2(x,y))$. The social welfare under the first-best mechanism M^* equals $x+1-y$.

By definition, the maximal competitive ratio of $LIP(a)$ can be computed as

$$\min_{1\geq x\geq y\geq 0} \frac{S(x,y)}{x+1-y}$$

We now show how to bound the above expression from both below and above.

Let N be a large positive integer. Let $h = \frac{1}{N}$ be the step size. Let the x_i be defined as $x_i = ih$ for $i = 0, 1, \ldots, N$. Similarly, let the y_i be defined as $y_i = ih$ for $i = 0, 1, \ldots, N$.

We have that

$$\min_{1\geq x\geq y\geq 0} \frac{S(x,y)}{x+1-y} \geq \min_{N>i\geq j\geq 0}\{ \min_{\substack{x_i+h\geq x\geq x_i \\ y_j+h\geq y\geq y_j}} \frac{S(x,y)}{x_i+h+1-y_j}\}$$

$$\geq \min_{N>i\geq j\geq 0} \frac{\begin{array}{c} x_ip_1(x_i, y_j+h) + (1-x_i-h)p_2(x_i+h, y_j) \\ +y_j(1-p_1(x_i+h, y_j)) \\ +(1-y_j-h)(1-p_2(x_i, y_j+h)) \end{array}}{x_i+h+1-y_j}$$

We also have that

$$\min_{1\geq x\geq y\geq 0} \frac{S(x,y)}{x+1-y} \leq \min_{N\geq i\geq j\geq 0} \frac{S(x_i,y_j)}{x_i+1-y_j}$$

$$= \min_{N\geq i\geq j\geq 0} \frac{\begin{array}{c} x_ip_1(x_i, y_j) + (1-x_i)p_2(x_i, y_j) \\ +y_j(1-p_1(x_i, y_j)) \\ +(1-y_j)(1-p_2(x_i, y_j)) \end{array}}{x_i+1-y_j}$$

We note that the x_i and the y_i are constants. The values of the $p_k(x_i, y_j)$ are also constants (for fixed a). That is, based on the above two inequalities, we are able to compute a constant upper bound and a constant lower bound on the maximal competitive ratio of $LIP(a)$. When $a = 2$, the lower bound is 0.828. Since any lower bound on the maximal competitive ratio is also a competitive ratio, $LIP(2)$ is (at least) 0.828-competitive. That is, the obtained $LIP(2)$ mechanism is near optimal for the case of two items (we recall that Theorem 10 says that any strategy-proof mechanism is at most 0.841-competitive).

Theorem 11. *For the case of two items and two agents, the competitive ratio of $LIP(2)$ is at least 0.828, and at most 0.829.*

Three or More Items

With more than two items, we need a different technique to bound the maximal competitive ratio of a given LIP mechanism.

Let α be the maximal competitive ratio of $LIP(a)$ (for some given a and m). Let $\mathbb{W}$ be the set of full ranked type vectors under $LIP(a)$. Let $\alpha^{\mathbb{W}}$ be the maximal competitive ratio of $LIP(a)$ if we restrict the type space to $\mathbb{W}$. The following proposition says that a lower bound on α can be obtained based on $\alpha^{\mathbb{W}}$.

Proposition 65. *Let α be the maximal competitive ratio of $LIP(a)$. Let $\alpha^{\mathbb{W}}$ be the maximal competitive ratio of $LIP(a)$ if we restrict the type space to the set of full ranked type vectors $\mathbb{W}$. We have*

$$\frac{a+b}{a+2mb}\alpha^{\mathbb{W}} \leq \alpha$$

Before proving this proposition, let us introduce the following definition and lemma.

Definition 11. Let $\vec{v} = (v_1, v_2, \ldots, v_m)$, which may or may not be full ranked. Let $\vec{v}$'s payment vector $(t_1, t_2, \ldots, t_m)$ be such that an agent with true type $\vec{v}$ will spend t_j units of artificial currency on item j (when she is the dictator). We define $\phi(\vec{v}) = (v'_1, v'_2, \ldots, v'_m)$, where $v'_j = \frac{at_j+b}{a+mb}$ for all j. That is, $\phi(\vec{v})$ is the (unique) full ranked type vector corresponding to the payment vector of $\vec{v}$.

If $\vec{v}$ is already full ranked, then $\phi(\vec{v}) = \vec{v}$. In any case, an agent with true type $\phi(\vec{v})$ will act in the same way as an agent with true type $\vec{v}$, since their corresponding payment vectors are the same.

Lemma 8. *$\forall \vec{v} = (v_1, v_2, \ldots, v_m), \forall j$, let $\phi(\vec{v}) = (v'_1, \ldots, v'_m)$. Then, we have $v_j + \frac{b}{a+mb} \geq v'_j$ and $v_j \frac{a+b}{a+mb} \leq v'_j$. That is, if we change $\vec{v}$ into $\phi(\vec{v})$, the value of an element increases at most by $\frac{b}{a+mb}$, and the value of an element decreases at most by a factor of $\frac{a+b}{a+mb}$.*

Proof. Let $(t_1, t_2, \ldots, t_m)$ be the payment vector of $\vec{v}$ and $\phi(\vec{v})$. Let $S = \{j | t_j > 0, j = 1, 2, \ldots, m\}$ and $T = \{j | t_j = 0, j = 1, 2, \ldots, m\}$. We have that for all $j \in S$, $\frac{v_j}{at_j+b} = C$ for a common constant C. We also have that for all $j \in T$, $C \geq \frac{v_j}{at_j+b} = \frac{v_j}{b}$.

We get $\sum_{j\in S} v_j = C(a+|S|b)$. We also get $\sum_{j\in T} v_j \leq C(|T|b)$. Since $\sum_{j\in S\cup T} v_j = 1$, we have $C(a+mb) \geq 1$. That is, for $j \in S$, $v_j \geq \frac{at_j+b}{a+mb} = v'_j$. For $j \in T$, $v'_j = \frac{b}{a+mb}$. Therefore, for any j, $v_j + \frac{b}{a+mb} \geq v'_j$.

Since $\sum_{j\in S\cup T} v_j = 1$ and $v_j \geq 0$ for all j, we have $\sum_{j\in S} v_j \leq 1$. That is, $C(a+b) \leq C(a+|S|b) \leq 1$. That is, $C \leq \frac{1}{a+b}$. Hence, for any j, $v_j \leq \frac{at_j+b}{a+b}$. Let us recall that $v'_j = \frac{at_j+b}{a+mb}$. Therefore, for any j, $v_j \frac{a+b}{a+mb} \leq v'_j$. □

Now we are ready to prove Proposition 65.

Proof of Proposition 65. Let $\vec{v_1}, \vec{v_2} \in \Omega$ be any two type vectors.

Let S be the obtained social welfare (under $LIP(a)$) when the agents report $\vec{v_1}$ and $\vec{v_2}$, respectively. Let M be the first-best social welfare when the agents report $\vec{v_1}$ and $\vec{v_2}$, respectively. Let S^ϕ be the obtained social welfare (under $LIP(a)$) when the agents report $\phi(\vec{v_1})$ and $\phi(\vec{v_2})$, respectively. Let M^ϕ be the first-best social welfare when the agents report $\phi(\vec{v_1})$ and $\phi(\vec{v_2})$, respectively.

We consider what happens when agents report $\phi(\vec{v_1})$ and $\phi(\vec{v_2})$ instead of $\vec{v_1}$ and $\vec{v_2}$. The allocation does not change. Since there are m items and by Lemma 8 the valuation of an item goes up by at most $\frac{b}{a+mb}$, we have $S^\phi \leq m\frac{b}{a+mb} + S$. Since by Lemma 8 the valuation of an item goes down by at most a factor of $\frac{a+b}{a+mb}$, we have $M^\phi \geq \frac{a+b}{a+mb}M$. Therefore $\frac{S+m\frac{b}{a+mb}}{\frac{a+b}{a+mb}M} \geq \frac{S^\phi}{M^\phi}$. Since $S \geq 1$ (as in the proof of Proposition 61), we have $\frac{S+m\frac{b}{a+mb}S}{\frac{a+b}{a+mb}M} \geq \frac{S^\phi}{M^\phi}$. That is, $\frac{S}{M} \geq \frac{a+b}{a+2mb}\frac{S^\phi}{M^\phi} \geq \frac{a+b}{a+2mb}\alpha^{\mathbb{W}}$. $\square$

Proposition 65 implies that if we can get a lower bound on $\alpha^{\mathbb{W}}$, then by multiplying it by $\frac{a+b}{a+2mb}$, we get a lower bound on α. So, we now focus on deriving a lower bound on the maximal competitive ratio of $LIP(a)$ considering only full ranked type vectors.

Let x,y be the agents' valuations for item 1 (or any other item). Without loss of generality, we assume $x \geq y$. Since we are only dealing with full ranked type vectors, we have $x = \frac{at_x+b}{a+mb}$ for some $0 \leq t_x \leq 1$, where t_x is the amount of artificial currency agent 1 spends on item 1 when she is the dictator. Similar observations hold for y. That is, $y = \frac{at_y+b}{a+mb}$ for some $0 \leq t_y \leq 1$, where t_y is the amount of artificial currency agent 2 spends on item 1 when she is the dictator. Let $u = \frac{y}{x}$. We have $\frac{b}{a+b} \leq u \leq 1$.

Under $LIP(a)$, the proportion of item 1 won by agent 1 when 1 is the dictator is $\frac{\ln(at_x+b)}{a} - \frac{\ln(b)}{a}$. The proportion of item 1 won by agent 1 when 1 is not the dictator is $1 - \frac{\ln(at_y+b)}{a} + \frac{\ln(b)}{a}$. In total, the proportion of item 1 won by agent 1 is $\frac{1}{2} + \frac{\ln(\frac{at_x+b}{at_y+b})}{2a} = \frac{1}{2} + \frac{\ln(\frac{x}{y})}{2a} = \frac{-\ln(u)}{2a} + \frac{1}{2}$. Similarly, the proportion of item 1 won by agent 2 is $\frac{\ln(u)}{2a} + \frac{1}{2}$.

We use $R(x,y)$ to denote the sum of the agents' utilities derived from item 1 when the agents' valuations for item 1 are x and y, respectively ($x \geq y$). Let $\theta(a)$ be defined as the minimum ratio between $R(x, y)$ and x over all x, y. That is, $\theta(a)$ is the minimum ratio of achieved utility over optimal utility for item 1 under $LIP(a)$, when we only consider full ranked vectors. $\theta(a)$ only depends on a (not on m). We call it the *intrinsic value* of a.

Proposition 66. *The intrinsic value $\theta(a)$ is less than or equal to the maximal competitive ratio of $LIP(a)$ considering only full ranked type vectors.*

Proof. By symmetry over the items, the achieved utility over optimal utility for any item is at least $\theta(a)$. Hence, the maximal competitive ratio is at least $\theta(a)$. $\square$

Let N be a large positive integer. Let $h = \frac{a}{N(a+b)}$ be the step size. Let the u_i be defined as $u_i = \frac{b}{a+b} + ih$ for $i = 0, 1, \ldots, N$.

We observe that

$$\theta(a) = \min_{\frac{b}{a+mb} \leq y \leq x \leq \frac{a+b}{a+mb}} \frac{x(\frac{-\ln(u)}{2a} + \frac{1}{2}) + y(\frac{\ln(u)}{2a} + \frac{1}{2})}{x}$$

$$= \min_{\frac{b}{a+b} \leq u \leq 1} \frac{-\ln(u)}{2a} + \frac{1}{2} + \frac{u\ln(u)}{2a} + \frac{u}{2}$$

$$\geq \min_{0 \leq i < N} \min_{u_i \leq u \leq u_i + h} \frac{(u-1)\ln(u)}{2a} + \frac{1}{2} + \frac{u}{2}$$

$$\geq \min_{0 \leq i < N} \frac{(u_i + h - 1)\ln(u_i + h)}{2a} + \frac{1}{2} + \frac{u_i}{2}$$

Given a, the u_i are constants. The above expression is the minimum of N constants. It gives a lower bound on $\theta(a)$. We denote it by $\underline{\theta(a)}$. The following expression gives an upper bound on $\theta(a)$ (denoted by $\overline{\theta(a)}$).

$$\theta(a) = \min_{\frac{b}{a+b} \leq u \leq 1} \frac{-\ln(u)}{2a} + \frac{1}{2} + \frac{u\ln(u)}{2a} + \frac{u}{2}$$

$$\leq \min_{0<i\leq N} \frac{(u_i-1)\ln(u_i)}{2a} + \frac{1}{2} + \frac{u_i}{2}$$

$$\leq \min_{0\leq i<N} \frac{(u_i+h-1)\ln(u_i+h)}{2a} + \frac{1}{2} + \frac{u_i}{2} + \frac{h}{2}$$

That is, the obtained lower bound $\underline{\theta(a)}$ and upper bound $\overline{\theta(a)}$ differ only by at most $\frac{h}{2}$, which can be made arbitrarily small.

Since $\theta(a) \leq \alpha^{\mathbb{W}}$, we have that α is bounded below by $\frac{a+b}{a+2mb}\theta(a)$.[14]

Next, we prove that $\theta(a)$ serves as an upper bound on α.[15]

Proposition 67. $\theta(a) \geq \alpha$.

Proof. Let $\bar{\alpha}$ be the maximal competitive ratio of $LIP(a)$ when there are only two items. We have $\bar{\alpha} \geq \alpha$. Hence we only need to show $\theta(a) \geq \bar{\alpha}$.

For the case of two items, let us consider the case where agent 1's type vector is $(\frac{u}{u+1}, \frac{1}{u+1})$, and agent 2's type vector is $(\frac{1}{u+1}, \frac{u}{u+1})$. Here, $\frac{b}{a+b} \leq u \leq 1$. It is easy to see that these two type vectors are full ranked. The utility of agent 1 under $LIP(a)$ equals $\frac{u}{u+1}(\frac{1}{2}+\frac{\ln(u)}{2a})+\frac{1}{u+1}(\frac{1}{2}+\frac{-\ln(u)}{2a})$. The utility of agent 2 is the same. The first-best social welfare is $\frac{2}{u+1}$. So, $\bar{\alpha}$ is at most $2\frac{\frac{1}{2}+\frac{u}{u+1}\frac{\ln(u)}{2a}+\frac{1}{u+1}\frac{-\ln(u)}{2a}}{\frac{2}{u+1}} = \frac{u+1}{2} + u\frac{\ln(u)}{2a} + \frac{-\ln(u)}{2a}$.

Since u can take any value from $\frac{b}{a+b}$ to 1, $\bar{\alpha} \leq \min_{\frac{b}{a+b}\leq u\leq 1} \frac{1}{2} + \frac{u}{2} + \frac{u\ln(u)}{2a} - \frac{\ln(u)}{2a}$. The expression on the right side of the inequality is exactly $\theta(a)$. □

Theorem 12 summarizes the development in this subsection.

Theorem 12. *For the case of m items and two agents, $LIP(a)$ is at least $\frac{a+b}{a+2mb}\theta(a)$-competitive, and at most $\theta(a)$-competitive.*

We illustrate the results in this subsection with Figure 3.4. For three to one hundred items, we searched for the LIP mechanism (from $\{LIP(a)|a = 0.01, 0.02, 0.03, \ldots,$

[14] When we compute this lower bound, we actually compute $\frac{a+b}{a+2mb}\underline{\theta(a)}$.

[15] When we compute this upper bound, we actually compute $\overline{\theta(a)}$.

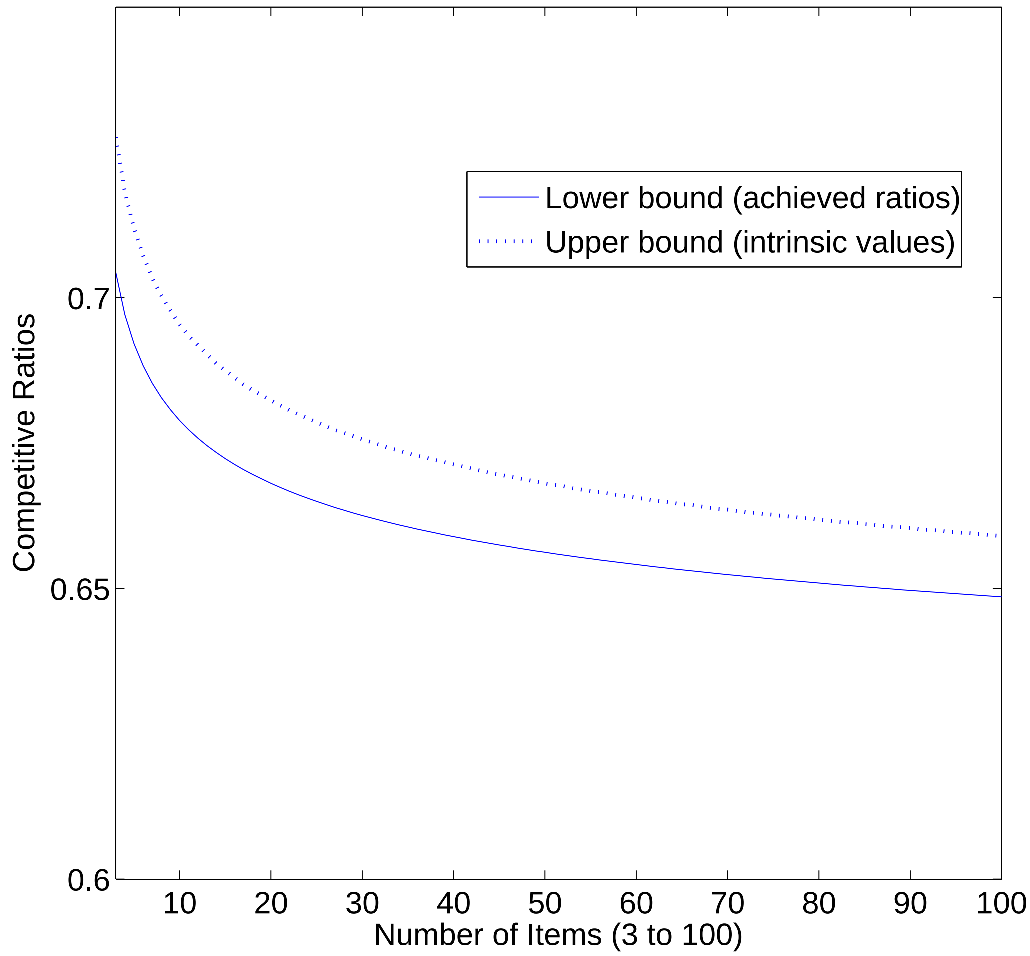

FIGURE 3.4: LIP mechanisms' competitive ratios

20}) that maximizes $\frac{a+b}{a+2mb}\theta(a)$ (the corresponding upper bounds $\theta(a)$ are also presented).

3.2.5 Large numbers of items

We now show a negative result: as the number of items goes to infinity, any increasing-price mechanism (whether it is linear or nonlinear) has maximal competitive ratio 0.5. That is, in the limit, they are no more competitive than the mechanism that simply divides the items evenly.

Theorem 13. *For the case of two agents, as the number of items m goes to infinity, the maximal competitive ratio of any increasing-price mechanism is* 0.5.

Proof. Let M be any increasing-price mechanism, characterized by the price function f. Let the type vectors of the agents be $(\frac{f(1)}{f(1)+(m-1)f(0)}, \frac{f(0)}{f(1)+(m-1)f(0)}, \ldots, \frac{f(0)}{f(1)+(m-1)f(0)})$ and $(1, 0, \ldots, 0)$, respectively. Either agent, when she is the dictator, will choose to spend all her artificial currency on item 1.

When agent 1 is the dictator, the social welfare under M equals $\frac{f(1)}{f(1)+(m-1)f(0)}$. When agent 2 is the dictator, the social welfare under M equals $1 + \frac{(m-1)f(0)}{f(1)+(m-1)f(0)}$. The social welfare under the first-best mechanism equals $1 + \frac{(m-1)f(0)}{f(1)+(m-1)f(0)}$. The competitive ratio of M is then at most $\frac{1}{1+\frac{(m-1)f(0)}{f(1)+(m-1)f(0)}} = \frac{f(1)+(m-1)f(0)}{f(1)+2(m-1)f(0)}$. As $m \to \infty$, this ratio goes to 0.5. That is, the maximal competitive ratio of any increasing-price mechanism is at most 0.5 as $m \to \infty$. On the other hand, 0.5 is a lower bound on the competitive ratios of strategy-proof mechanisms by Proposition 61. □

3.3 Summary

In this chapter, we applied CFAMD to the problem of designing resource allocation mechanisms that do not rely on payments at all. This is useful in settings where no currency has (yet) been established (as may be the case, for example, in a peer-to-peer network, as well as in many other multiagent systems); or where payments are prohibited by law; or where payments are otherwise inconvenient. In Section 3.1, we studied the problem of allocating a single item repeatedly among multiple competing agents. We introduced an artificial payment system, which enabled us to construct repeated allocation mechanisms without payments based on one-shot allocation mechanisms with payments. Under certain restrictions on the discount factor, we proposed several (Bayes-Nash) incentive compatible repeated allocation mechanisms based on artificial payments. We proved that our mechanisms are competitive against the first-best allocation. In Section 3.2, we investigated the problem of allocating multiple items among two competing agents in a single-round prior-free setting. We introduced the family of linear increasing-price (LIP) mechanisms. The LIP mechanisms are strategy-proof and only rely on artificial payments. We showed how to solve for mechanisms within the LIP family that are competitive against the first-best allocation. In both scenarios discussed in this chapter, the first-best allocation can be obtained by mechanisms with payments. Our results imply that in some cases, artificial payments may be used to replace real monetary payments, without incurring too much loss in social welfare.

4

False-name-proofness with Bid Withdrawal

With the rapid development of electronic commerce, Internet auctions have become increasingly popular over the years [83, 107, 97]. Unlike traditional auctions, typical Internet auctions pose no geographical constraint. That is, sellers and bidders from all over the world can participate in an Internet auction remotely over the Internet, without having to physically attend the auction event. For sellers, this reduces the cost of running an auction. For bidders, this lowers the entry cost. Effectively, in an individually rational auction mechanism (a mechanism that guarantees nonnegative utilities for the agents), a bidder loses nothing (but time) by participating in an auction. On the one hand, this encourages more bidders to join the auction, which potentially leads to higher revenue for the seller, as well as a higher efficiency for the bidders. On the other hand, it enables the bidders to manipulate by submitting multiple bids via multiple fictitious identities (*e.g.*, user accounts linked to different e-mail addresses).

The line of research on preventing manipulation via multiple fictitious identities in Internet auctions was explicitly framed by the groundbreaking work of Yokoo *et al.* [111]. Extending strategy-proofness, the authors define an auction mechanism to be *false-name-proof* if the mechanism is not only strategy-proof, but also, under this mechanism, an agent cannot benefit from submitting multiple bids under false names (fictitious identities). The authors also extended the revelation principle to incorporate false-name-proofness. That is (roughly stated), in settings where false-name bids are possible, it is without loss of generality to focus only on false-name-proof mechanisms.

Several false-name-proof mechanisms have been proposed for general combinatorial auction settings. These are the Set mechanism [108], the Minimal Bundle (MB) mechanism [108], and the Leveled Division Set (LDS) mechanism [110].[1] Other work on false-name-proofness includes the following. For general combinatorial auction settings, Yokoo [108] and Todo *et al.* [102] characterized the payment rules and the allocation rules of false-name-proof mechanisms, respectively. False-name proofness has also been studied in the context of voting mechanisms [29, 106]. Finally, Conitzer [28] proposed the idea of preventing false-name manipulation by verifying the identities of certain limited subsets of agents.

Focusing primarily on combinatorial auctions, this chapter continues the line of research on false-name-proofness by considering an even more powerful variant of false-name manipulation: an agent can submit multiple false-name bids, but then, once the allocation and payments have been decided, withdraw some of her false-name identities (have some of her false-name identities refuse to pay). While these withdrawn identities will not obtain the items they won, their initial presence may have been beneficial to the agent's other identities, as shown in the following example:

Example 18. There are three single-minded agents $1, 2, 3$ and two items A, B. Agent 1 bids 4 on $\{A, B\}$. Agent 2 bids 2 on $\{B\}$. Let us analyze the strategic options for agent 3, who is single-minded on $\{A\}$, with valuation 1. (That is, $\forall S \subseteq \{A, B\}$, agent

[1] A very recent paper [69] introduces a new mechanism called the ARP mechanism. However, this mechanism requires the additional restriction that agents are single-minded.

3's valuation for S is 1 if and only if $\{A\} \subseteq S$.) The mechanism under consideration is the VCG mechanism.

If agent 3 reports truthfully, then she wins nothing and pays nothing. Her resulting utility equals 0.

If agent 3 attempts "traditional" false-name manipulation, that is, submitting multiple false-name bids, and honoring all of them at the end, then her utility is still at most 0: if 3 wins both items with one identity, then she has to pay at least 4 (while her valuation for the items is only 1); if 3 wins both items with two identities (one item for each identity), then the identity winning $\{B\}$ has to pay at least 2; if 3 wins only $\{B\}$ or nothing, then her utility is at most 0; if 3 wins only $\{A\}$ (in which case $\{B\}$ has to be won by agent 2), then 3's winning identity's payment equals the other identities' overall valuation for $\{A, B\}$ (at least 4), minus 2's valuation for $\{B\}$ (which equals 2). That is, in this case, 3 has to pay at least 2. So, overall, 3's utility is at most 0 if she honors all her bids.

However, agent 3 can actually benefit from submitting multiple false-name bids, as long as she can withdraw some of them. For example, 3 can use two identities, $3a$ and $3b$. $3a$ bids 1 on $\{A\}$. $3b$ bids 4 on $\{B\}$. At the end, $3a$ wins $\{A\}$ for free, and $3b$ wins $\{B\}$ for 2. If 3 can withdraw identity $3b$ (*e.g.*, by never checking that e-mail account anymore), never making the payment and never collecting $\{B\}$, then, she has obtained $\{A\}$ for free, resulting in a utility of 1.

If we wish to guard against manipulations like the above, we need to extend the false-name-proofness condition. We refer to the new condition as *false-name-proofness with withdrawal (FNPW)*. It requires that, regardless of what other agents do, an agent's optimal strategy is to report truthfully using a single identity, even if she has the option to submit multiple false-name bids, and withdraw some of them at the end of the auction.

To our knowledge, this stronger version of false-name-proofness has not previously been considered. Whether it is more or less reasonable than the original version depends on the context. For example, in an auction, it may be possible to require each participant to place the amount of her bid in escrow, which would prevent manipulation based on withdrawal. However, in some auction contexts, such an arrangement would be too unattractive to the bidders; it also reduces the anonymity of bidding. Additionally, if we are in a setting where the payments are not monetary, but rather are in terms of performance of future services, then it is not possible to put the payments in escrow.

In any case, FNPW is a useful conceptual tool for analyzing false-name-proof mechanisms. Indeed, this chapter also contributes to the research on false-name-proofness in the traditional sense. Since FNPW is stronger than FNP, the mechanisms we propose in this chapter, as well as the automated mechanism design technique, should be of interest in the FNP context as well.

The chapter is organized as follows. In Section 4.1, we formalize the problem we study. In Section 4.2, we give a sufficient and necessary condition on the type space for the VCG mechanism to be FNPW. In Section 4.3, we characterize both the payment rules and the allocation rules of FNPW mechanisms in general combinatorial auctions. We also derive a sufficient condition that can be used to check whether a mechanism is FNPW. In Section 4.4, we propose the *maximum marginal value item pricing (MMVIP)* mechanism, which we prove is FNPW. In Section 4.5, we propose an automated mechanism design technique that transforms any feasible mechanism into an FNPW mechanism. This technique builds on the sufficient condition in Section 4.3. In Section 4.6, we show that, under a minor condition, the mechanism that sells all the items as a single bundle has the highest worst-case efficiency ratio among all FNPW mechanisms. Finally, in Section 4.7, we give a characterization of FNP(W) social choice rules.

4.1 Formalization

We will still use the following standard notation:

- $I = \{1, 2, \ldots, n\}$: the set of agents
- $G = \{1, 2, \ldots, m\}$: the set of items
- Θ: the type space of each agent
- $\theta_i \in \Theta$: agent i's reported type (since we consider only strategy-proof mechanisms, when there is no ambiguity, we also use θ_i to denote i's true type)
- $-i$: the set of agents other than agent i
- $\theta_{-i} \in \Theta^{n-1}$: types reported by agents other than agent i

We study combinatorial auction settings satisfying the following assumptions:

- $\forall \theta \in \Theta$, we have $v(\theta, \emptyset) = 0$.
- $\forall B_1 \subseteq B_2 \subseteq G$, $\forall \theta \in \Theta$, we have $v(\theta, B_1) \leq v(\theta, B_2)$. That is, there is *free disposal*.
- An agent can have any valuation function satisfying the above conditions. That is, we are dealing with *rich domains* [13]. It should be noted that in Section 4.2, we study how restrictive the type space has to be in order for the VCG mechanism to be FNPW. That is, we do not have the rich-domain assumption in Section 4.2, which is an exception.

A mechanism consists of an allocation rule $X : (\Theta, \Theta^{n-1}) \to \mathcal{P}(G)$ and a payment rule $P : (\Theta, \Theta^{n-1}) \to \mathbb{R}$. $X(\theta_i, \theta_{-i})$ is the bundle agent i receives when reporting θ_i (when the other agents report θ_{-i}). $P(\theta_i, \theta_{-i})$ is the payment agent i has to make when reporting θ_i (when the other agents report θ_{-i}). When there is no ambiguity about the other agents' types, we simply use $X(\theta_i)$ and $P(\theta_i)$ in place of $X(\theta_i, \theta_{-i})$ and $P(\theta_i, \theta_{-i})$.

Throughout the chapter, we only consider mechanisms satisfying the following conditions:

- *Strategy-proofness:* $\forall \theta_i, \theta'_i, \theta_{-i}$, we have $v(\theta_i, X(\theta_i)) - P(\theta_i) \geq v(\theta_i, X(\theta'_i)) - P(\theta'_i)$. That is, if an agent uses only one identity, then truthful reporting is a dominant strategy.

- *Pay-only:* $\forall \theta_i, \theta_{-i}$, we have $P(\theta_i) \geq 0$.

- *(Ex post) individual rationality:* $\forall \theta_i, \theta_{-i}$, we have $v(\theta_i, X(\theta_i)) - P(\theta_i) \geq 0$. That is, if an agent reports truthfully, then her utility is guaranteed to be nonnegative. This condition also implies that if an agent does not win any items, or has valuation 0 for all the items, then her payment must be 0.

- *Consumer sovereignty:* $\forall \theta_{-i}, \forall B \subseteq G$, there exists $\theta_i \in \Theta$ such that $X(\theta_i, \theta_{-i}) \supseteq B$. That is, no matter what the other agents bid, an agent can always win any bundle (possibly at the cost of a large payment).

- *Determinism and symmetry:* We only consider deterministic mechanisms that are symmetric over both the agents and the items (except for ties).

Yokoo [108] showed that in our setting, the mechanisms satisfying the above conditions coincide with the *(anonymous) price-oriented, rationing-free (PORF)* mechanisms. Similar price-based representations have also been presented by others, including [75]. The PORF mechanisms work as follows:

- The agents submit their reported types.

- The mechanism is characterized by a price function $\chi : \mathcal{P}(G) \times \Theta^{n-1} \to [0, \infty)$. For any agent i, for any multi-set θ_{-i} of types reported by the other agents, for any set of items $S \subseteq G$, $\chi(S, \theta_{-i})$ is the price of S offered to i by the mechanism. That is, i can purchase S at a price of $\chi(S, \theta_{-i})$. $\forall \theta_{-i}$, we have $\chi(\emptyset, \theta_{-i}) = 0$. That is, the price of nothing is always zero. $\forall \theta_{-i}$, $\forall S_1 \subseteq S_2 \subseteq G$, we have $\chi(S_1, \theta_{-i}) \leq \chi(S_2, \theta_{-i})$. That is, a larger bundle always has a higher (or the same) price.

- The mechanism will select a bundle for agent i that is optimal for her given the prices, that is, the bundle chosen for i is in

$$\arg\max_{S \subseteq G} \{v(\theta_i, S) - \chi(S, \theta_{-i})\}$$

  The agent then pays the price for this bundle.

- Naturally, the mechanism must ensure that no item is allocated to two different agents. This involves setting prices carefully, as well as breaking ties.

Since all *feasible* mechanisms (mechanisms that satisfy the desirable conditions in our setting) are PORF mechanisms, besides using X (the allocation rule) and P (the payment rule) to refer to a mechanism, we can also use the price function χ to refer to a mechanism, namely, the PORF mechanism with price function χ.[2]

In the remainder of this section, we formally define the traditional false-name-proofness (FNP) condition, as well as our new false-name-proofness with withdrawal (FNPW) condition.

Definition 12. FNP. A mechanism characterized by allocation rule X and payment rule P is FNP if and only if it satisfies the following:

[2] Technically, there can be multiple PORF mechanisms with the same price function due to tie-breaking, but this will generally not be an issue.

$$\forall \theta_i,\ \forall \theta_{i1}, \theta_{i2}, \ldots, \theta_{ik},\ \forall \theta_{-i}, \text{ we have}$$
$$v(\theta_i, X(\theta_i, \theta_{-i})) - P(\theta_i, \theta_{-i}) \geq$$
$$v(\theta_i, \bigcup_{j=1}^{k} X(\theta_{ij}, \theta_{-i} \cup (\bigcup_{t \neq j} \theta_{it}))) - \sum_{j=1}^{k} P(\theta_{ij}, \theta_{-i} \cup (\bigcup_{t \neq j} \theta_{it}))$$

That is, truthful reporting using a single identifier is always better than submitting multiple false-name bids.

Definition 13. FNPW. A mechanism characterized by allocation rule X and payment rule P is FNPW if and only if it satisfies the following:

$$\forall \theta_i,\ \forall \theta_{i1}, \theta_{i2}, \ldots, \theta_{ik},\ \forall \theta'_{i1}, \theta'_{i2}, \ldots, \theta'_{iq},\ \forall \theta_{-i}, \text{ we have}$$
$$v(\theta_i, X(\theta_i, \theta_{-i})) - P(\theta_i, \theta_{-i}) \geq$$
$$v(\theta_i, \bigcup_{j=1}^{k} X(\theta_{ij}, \theta_{-i} \cup (\bigcup_{t \neq j} \theta_{it}) \cup (\bigcup \theta'_{it}))) - \sum_{j=1}^{k} P(\theta_{ij}, \theta_{-i} \cup (\bigcup_{t \neq j} \theta_{it}) \cup (\bigcup \theta'_{it}))$$

That is, truthful reporting using a single identifier is always better than submitting multiple false-name bids and then withdrawing some of them.

Actually, FNPW is exactly equivalent to FNP plus the following condition:

Definition 14. Others' bids do not help (OBDNH). A mechanism characterized by allocation rule X and payment rule P satisfies the OBDNH condition if and only if

$$\forall \theta_i,\ \forall \theta',\ \forall \theta_{-i}, \text{ we have}$$
$$v(\theta_i, X(\theta_i, \theta_{-i})) - P(\theta_i, \theta_{-i}) \geq v(\theta_i, X(\theta_i, \theta_{-i} \cup \theta')) - P(\theta_i, \theta_{-i} \cup \theta')$$

That is, an agent's utility for reporting truthfully does not increase if we add another agent.

Theorem 14. *FNPW is equivalent to FNP plus OBDNH.*

Proof. We first prove that FNPW implies FNP and OBDNH.

It is straightforward that FNPW implies FNP. We only need to prove that FNPW implies OBDNH. $\forall\theta_i$, $\forall\theta_{-i}$, $\forall\theta'$, let $k = 1$, $\theta_{i1} = \theta_i$, $q = 1$, and $\theta'_{i1} = \theta'$. With these assignments, the FNPW condition reduces to the OBDNH condition.

We now prove that FNP and OBDNH together imply FNPW. $\forall\theta_i$, $\forall\theta'_{i1}, \theta'_{i2}, \ldots, \theta'_{iq}$, $\forall\theta_{-i}$, according to OBDNH, we have $v(\theta_i, X(\theta_i, \theta_{-i})) - P(\theta_i, \theta_{-i}) \geq v(\theta_i, X(\theta_i, \theta_{-i} \cup (\bigcup \theta'_{it}))) - P(\theta_i, \theta_{-i} \cup (\bigcup \theta'_{it}))$. Then, according to FNP, $\forall\theta_{i1}, \theta_{i2}, \ldots, \theta_{ik}$, replacing θ_{-i} by $\theta_{-i} \cup (\bigcup \theta'_{it})$, we obtain $v(\theta_i, X(\theta_i, \theta_{-i} \cup (\bigcup \theta'_{it}))) - P(\theta_i, \theta_{-i} \cup (\bigcup \theta'_{it})) \geq v(\theta_i, \bigcup_{j=1}^{k} X(\theta_{ij}, \theta_{-i} \cup (\bigcup_{t \neq j} \theta_{it}) \cup (\bigcup \theta'_{it}))) - \sum_{j=1}^{k} P(\theta_{ij}, \theta_{-i} \cup (\bigcup_{t \neq j} \theta_{it}) \cup (\bigcup \theta'_{it}))$. Combining the inequalities, we obtain exactly the FNPW condition. □

According to Theorem 14, to check whether an FNP mechanism is FNPW, we only need to check whether it satisfies OBDNH.

Proposition 68. *The Leveled Division Set (LDS) mechanism [110] does not satisfy OBDNH. That is, LDS is not FNPW in general.*[3]

The general LDS mechanism is rather complicated. Instead of describing LDS in its general form, we focus on a specific LDS mechanism for three items, which is characterized by reserve price 1 and the following two levels:

$$[\{(A, B, C)\}] \text{ and } [\{(A, B), (C)\}, \{(A), (B, C)\}]$$

The mechanism works as follows. If there are at least two agents whose valuations for $\{A, B, C\}$ are at least 3, then we combine $\{A, B, C\}$ into one bundle, and run the Vickrey auction. If every agent's valuation for $\{A, B, C\}$ is less than 3, then we do the following. We first introduce a dummy agent into the system. The dummy agent has an additive valuation function and values every item at 1. We only allow

[3] We will show later that the other two known FNP mechanisms, that is, the Set mechanism [108] and the Minimal Bundle mechanism [108], are both FNPW.

five types of allocations: 1) The dummy agent wins everything. 2) The dummy agent wins one of $\{A, B\}$ and $\{C\}$, and a non-dummy agent wins the other. 3) The dummy agent wins one of $\{A\}$ and $\{B, C\}$, and a non-dummy agent wins the other. 4) A non-dummy agent wins one of $\{A, B\}$ and $\{C\}$, and another non-dummy agent wins the other. 5) A non-dummy agent wins one of $\{A\}$ and $\{B, C\}$, and another non-dummy agent wins the other. We run the VCG mechanism on this restricted set of possible allocations. Finally, if there is only one agent whose valuation for $\{A, B, C\}$ is at least 3, then this agent is the only winner. She has the option to purchase all the items at price 3, or to obtain the result she would have obtained if everyone (including the dummy agent) were to join in the above maximal-in-range mechanism.

Proof. We only need to prove that the above specific LDS mechanism does not satisfy OBDNH. We consider the following scenario. There are two agents. Agent 1 bids 2.2 on $\{A, B\}$. Agent 2 is single-minded, valuing $\{A\}$ at 1.1. Under the above LDS mechanism, if 2 reports truthfully, then $\{A, B\}$ is allocated to 1, and $\{C\}$ is allocated to the dummy agent (thrown away). That is, if 2 reports truthfully, then her utility equals 0. If, besides 2's true identity, 2 also submits a false-name bid of 2.9 on $\{B, C\}$, then $\{B, C\}$ will be allocated to 2's false-name identity (2 will withdraw this identity, that is, refuse to pay for this bundle), and $\{A\}$ will be allocated to 2's true identity at a price of 1. That is, 2 now has utility 0.1. We conclude that, in general, LDS does not satisfy OBDNH, and hence is not FNPW. □

4.2 Restriction on the type space so that VCG is FNPW

The VCG mechanism [103, 25, 52] satisfies several nice properties, including efficiency, strategy-proofness, individual rationality, and the non-deficit property. Unfortunately, as shown by Yokoo *et al.* [111], the VCG mechanism is not FNP for general type spaces. One sufficient condition on the type space for the VCG mechanism to be FNP is as follows:

Definition 15. Submodularity [111]. For any set of bidders Y, whose types are drawn from Θ, $\forall S_1, S_2 \subseteq G$, we have $U(S_1, Y) + U(S_2, Y) \geq U(S_1 \cup S_2, Y) + U(S_1 \cap S_2, Y)$. Here, $U(S, Y)$ is defined as the total utility of bidders in Y, if we allocate items in S to these bidders efficiently.

That is, if the type space Θ satisfies the above condition, then the VCG mechanism is FNP. In this section, we aim to characterize type spaces for which VCG is FNPW. We consider restricted type spaces (that make the VCG mechanism FNPW) in this section. In other sections, unless specified, we assume that the rich-domain condition holds.

Theorem 15. *If the type space satisfies the submodularity condition, then the VCG mechanism is FNPW. Conversely, if the mechanism is FNPW, and additionally the type space contains the additive valuations, then the type space satisfies the submodularity condition.*

That is, submodularity does not only imply FNP, it actually implies FNPW. Moreover, unlike for FNP, in the case of FNPW, the converse also holds—if we allow the additive valuations (those valuations which value any set of items at the sum of the values of its elements, with no complementarity and no substitutability).

Proof. We first prove that if the type space satisfies submodularity, then the VCG mechanism is FNPW. We consider agent i. Let K be the set of false-name identities i submits and keeps at the end. Let W be the set of false-name identities i submits and withdraws. We already know that submodularity is sufficient for the VCG mechanism to be FNP. Hence, if K contains multiple identities, then i might as well replace all of them by one identity that reports i's true type. We then show that the identities in W do not help i (OBDNH). We use S to denote the set of items won by i at the end. To win S, i pays the VCG price $U(G, \{-i\} \cup W) - U(G - S, \{-i\} \cup W)$ ($\{-i\}$ is the set of other agents). We use S' to denote the set of items won by identities in W, when we allocate items in $G - S$ to identities in $\{-i\} \cup W$ efficiently. We have that $U(G, \{-i\} \cup W) - U(G - S, \{-i\} \cup W) = U(G, \{-i\} \cup W) - U(G - S - S', \{-i\}) - U(S', W) \geq U(G - S', \{-i\}) + U(S', W) - U(G - S - S', \{-i\}) - U(S', W) = U(G - S', \{-i\}) - U(G - S - S', \{-i\})$. The submodularity condition implies that $U(G - S', \{-i\}) - U(G - S - S', \{-i\}) \geq U(G, \{-i\}) - U(G - S, \{-i\})$. But, the expression on the right-hand side of the inequality is the price i would be charged for S when she uses a single identifier. That is, i does not benefit from the false-name identities in W. Therefore, the VCG mechanism is FNPW if the type space satisfies submodularity.

Next, we prove that if the VCG mechanism is FNPW, then the type space must satisfy submodularity (if it contains the additive valuations). Let S be an arbitrary set of items. Let i be an agent that is interested in S. Since we allow additive valuations, such i always exists (*e.g.*, i may have a very large valuation for every item in S). If i bids truthfully, then she can win S at a price of $U(G, \{-i\}) - U(G - S, \{-i\})$. Let S' be another arbitrary set of items that does not intersect with S. For each item j in S', we introduce a false-name identity that is only interested in item j, with value c, where c is set to a very large value (*e.g.*, larger than $U(G, \{-i\})$). These false-name identities are allowed since we assume the type space contains the additive valuations. Let W be the set of identities introduced. With W, i can win S at a price of $U(G, \{-i\} \cup W) - U(G - S, \{-i\} \cup W)$. We have that $U(G, \{-i\} \cup W) - U(G - S, \{-i\} \cup W) = U(G - S', \{-i\}) + U(S', W) - U(G - S - S', \{-i\}) - U(S', W) = U(G - S', \{-i\}) - U(G - S - S', \{-i\})$. The new price should never be smaller than the old price. Otherwise, there is an incentive for i to submit false-name bids and withdraw them. That is, we have $U(G, \{-i\}) - U(G - S, \{-i\}) \leq$

$U(G-S', \{-i\}) - U(G-S-S', \{-i\})$. Let $S_1 = G-S$, $S_2 = G-S'$, and $Y = \{-i\}$. We have $U(S_1 \cap S_2, Y) - U(S_1, Y) \le U(S_2, Y) - U(S_1 \cup S_2, Y)$. Since S_1, S_2, and Y are arbitrary, we have submodularity. □

4.3 Characterization of FNPW mechanisms

Yokoo [108] and Todo *et al.* [102] characterized the payment rules (the price functions in the PORF representation) and the allocation rules of FNP mechanisms, respectively. In this section, we present similar results on the characterization of FNPW mechanisms.

4.3.1 Characterizing FNPW payments

We recall that in our setting, a feasible mechanism corresponds to a PORF mechanism, characterized by a price function χ. Yokoo [108] gave the following sufficient and necessary condition on χ for the mechanism characterized by χ to be FNP.

Definition 16. No superadditive price increase (NSA). Let O be an arbitrary set of agents.[4] We run mechanism χ (a PORF mechanism characterized by price function χ) for the agents in O. Let Y be an arbitrary subset of O. Let B_i ($i \in Y$) be the set of items agent i obtains. We must have

$$\sum_{i \in Y} \chi(B_i, O - \{i\}) \ge \chi(\bigcup_{i \in Y} B_i, O - Y).$$

By modifying the NSA condition, we get the following sufficient and necessary condition on χ for mechanism χ to be FNPW.

Definition 17. No superadditive price increase with withdrawal (NSAW). Let O be an arbitrary set of agents. We run mechanism χ for the agents in O. Let

[4] In a slight abuse of language, we also use "a set of agents" to refer to the types reported by this set of agents.

Y and Z be two arbitrary nonintersecting subsets of O. Let B_i ($i \in Y$) be the set of items agent i obtains. We must have

$$\sum_{i \in Y} \chi(B_i, O - \{i\}) \geq \chi(\bigcup_{i \in Y} B_i, O - Y - Z).$$

NSAW is equivalent to NSA plus the following condition.

Definition 18. Prices increase with agents (PIA). Let O be an arbitrary set of agents. Let a be another agent not in O. $\forall S \subseteq G$, we must have

$$\chi(S, O \cup \{a\}) \geq \chi(S, O).$$

That is, from the perspective of agent i, if another agent joins in, then the price i faces for any set of items must (weakly) increase.

Proposition 69. *NSAW is equivalent to NSA plus PIA.*

Proof. We first prove that NSAW implies NSA and PIA. It is straightforward that NSAW implies NSA, so we only need to show that NSAW implies PIA. Let R be an arbitrary set of agents. Let a be another agent not in R. $\forall S \subseteq G$, we can construct an agent (denoted by y) that wins S if we run χ on the agents in $R \cup \{a\} \cup \{y\}$ (*e.g.*, let y be single-minded on S, with a very large value). Let $Y = \{y\}$, $Z = \{a\}$, and $O = R \cup Y \cup Z$. NSAW implies that $\chi(S, R \cup Z) = \chi(S, R \cup \{a\}) \geq \chi(S, R)$. That is, NSAW implies PIA.

We now prove that NSA and PIA imply NSAW. PIA implies that $\chi(\bigcup_{i \in Y} B_i, O - Y - Z) \leq \chi(\bigcup_{i \in Y} B_i, O - Y)$. NSA implies that $\sum_{i \in Y} \chi(B_i, O - \{i\}) \geq \chi(\bigcup_{i \in Y} B_i, O - Y)$. Combining the two inequalities, we obtain the NSAW condition. □

Theorem 16. *Mechanism χ is FNPW if and only if χ satisfies the NSAW condition.*

Proof. We first prove that if χ satisfies NSAW, then the mechanism is FNPW. Let us consider a specific agent x. Let $O - Y - Z$ be the set of agents other than herself. Let Y be the set of false-name identities x submits and keeps at the end. Let Z be the set of false-name identities x submits but withdraws at the end. So, O is the set of all the identities. The set of items x receives at the end is $\bigcup_{i \in Y} B_i$, where B_i is the bundle won by identity i. The total price x pays is $\sum_{i \in Y} \chi(B_i, O - \{i\})$. According to NSAW, this price is at least $\chi(\bigcup_{i \in Y} B_i, O - Y - Z)$. That is, x would not be any worse off if she just used a single identity to buy $\bigcup_{i \in Y} B_i$. When x uses only one identity, her optimal strategy is to report truthfully. Therefore, if NSAW is satisfied, mechanism χ is FNPW.

Next, we prove that if mechanism χ is FNPW, then χ must satisfy NSAW. Suppose not, that is, suppose there exists some χ that corresponds to an FNPW mechanism, and there exist three nonintersecting sets of agents Y, Z, and $O - Y - Z$, such that $\sum_{i \in Y} \chi(B_i, O - \{i\}) < \chi(\bigcup_{i \in Y} B_i, O - Y - Z)$, where B_i is the bundle agent i obtains (when we apply mechanism χ to the agents in O). Let us consider a single-minded agent x, who values $\bigcup_{i \in Y} B_i$ at exactly $\chi(\bigcup_{i \in Y} B_i, O - Y - Z)$. If the set of other agents faced by x is $O - Y - Z$, then x has utility 0 if she reports truthfully using a single identifier. However, if x instead submits multiple false-name identities $Y + Z$, keeps those in Y and withdraws those in Z, then she will obtain her desired items at a lower price and end up with positive utility, contradicting the assumption that χ is FNPW. That is, if NSAW is not satisfied, then χ is not FNPW. □

4.3.2 A sufficient condition for FNPW

The NSAW condition in Subsection 4.3.1 leads to the following sufficient condition for mechanism χ to be FNPW.

Definition 19. Sufficient condition for no superadditive price increase with withdrawal (S-NSAW). Let O be an arbitrary set of agents. S-NSAW holds if we have both of the following conditions:

- **Discounts for larger bundles (DLB).** $\forall S_1, S_2 \subseteq G$ with $S_1 \cap S_2 = \emptyset$, $\chi(S_1, O) + \chi(S_2, O) \geq \chi(S_1 \cup S_2, O)$. That is, the sum of the prices of two disjoint sets of items must be at least the price of the joint set.

- **Prices increase with agents (PIA).**[5] $\forall S \subseteq G$, for any agent a that is not in O, $\chi(S, O \cup \{a\}) \geq \chi(S, O)$.

Proposition 70. *Mechanism χ is FNPW if χ satisfies S-NSAW.*

Proof. We only need to show that S-NSAW is stronger than NSAW (by Theorem 16, NSAW is sufficient (and necessary) for χ to be FNPW). Let χ satisfy S-NSAW. Let O be an arbitrary set of agents. We run mechanism χ on the agents in O. We divide O into three subgroups, Y, Z, and $O - Y - Z$. For $i \in Y$, let B_i be the bundle agent i obtains. By PIA, we have $\sum_{i \in Y} \chi(B_i, O - \{i\}) \geq \sum_{i \in Y} \chi(B_i, O - Y - Z)$. By DLB, we have $\sum_{i \in Y} \chi(B_i, O - Y - Z) \geq \chi(\bigcup_{i \in Y} B_i, O - Y - Z)$. Combining these inequalities, we can conclude that S-NSAW implies NSAW. □

S-NSAW is a cleaner, but more restrictive condition than NSAW. (To see why, note that even if DLB does not hold, NSA may still hold: even if $\chi(S_1, O) + \chi(S_2, O) < \chi(S_1 \cup S_2, O)$, it may be the case that by putting separate bids on S_1 and S_2, each of these bids makes the price for the other bundle go up, so that the result is still more expensive than buying $S_1 \cup S_2$ as a single bundle.) We find it easier to use S-NSAW to prove that a mechanism is FNPW (rather than using the more complex NSAW condition).[6] Let us recall the three existing FNP mechanisms (for general

[5] This is the same PIA condition as the one in Subsection 4.3.1.

[6] However, S-NSAW cannot be used to prove that a mechanism is *not* FNPW, because it is a more restrictive condition.

combinatorial auction settings): the Set mechanism, the MB Mechanism, and the LDS mechanism. We have already shown that LDS is not FNPW. With the help of S-NSAW, we can prove that both Set and MB are FNPW.

Proposition 71. *Both the Set mechanism and the MB mechanism satisfy the S-NSAW condition. Hence, they are FNPW.*

The Set mechanism simply combines all the items into a grand bundle. The grand bundle is then sold in a Vickrey auction. The MB (Minimal Bundle) mechanism builds on the concept of minimal bundles. A set of items S ($\emptyset \subsetneq S \subset G$) is called a *minimal bundle* for agent i if and only if $\forall S' \subsetneq S$, $v(i,S) > v(i,S')$. Under the MB mechanism, the price of a bundle S an agent faces is equal to the highest valuation value of a bundle, which is minimal and conflicting with S. Generally, MB coincides with Set, because usually the grand bundle is a minimal bundle for every agent (any smaller bundle usually gives at least slightly lower utility). The proof of the above proposition is straightforward.

We will also use S-NSAW to prove that the MMVIP mechanism that we propose (Section 4.4) is FNPW. The automated mechanism design technique for generating FNPW mechanisms (Section 4.5) is also based on S-NSAW.

4.3.3 Characterizing FNPW allocations

Todo *et al.* [102] gave the following characterization of the allocation rules of FNP mechanisms. We recall that $X(\theta_i, \theta_{-i})$ is the set of items that agent i wins if her reported type is θ_i and the reported types of the other agents are θ_{-i}. To simplify notation, we use $X(\theta_i)$ in place of $X(\theta_i, \theta_{-i})$ when there is no risk of ambiguity.

Definition 20. Weak-monotonicity [13]. X is weakly monotone if $\forall \theta_i, \theta_i', \theta_{-i}$, we have

$$v(\theta_i, X(\theta_i)) - v(\theta_i, X(\theta_i')) \geq v(\theta_i', X(\theta_i)) - v(\theta_i', X(\theta_i')).$$

Definition 21. Sub-additivity [102]. $\forall\theta_i$, $\forall\theta'_i$, $\forall\theta_{i1},\theta_{i2},\ldots,\theta_{ik}$, $\forall\theta'_{i1},\theta'_{i2},\ldots,\theta'_{ik}$, $\forall\theta_{-i}$, we have the following:

$$X(\theta_i) = \bigcup_{l=1}^{k} X_{+I^k_{-l}}(\theta_{il})$$

$$v(\theta'_i, X(\theta'_i)) = 0$$

$$X_{+I^k_{-l}}(\theta'_{il}) \supseteq X_{+I^k_{-l}}(\theta_{il})$$

$$v(\theta'_{il}, X_{+I^k_{-l}}(\theta'_{il})) = v(\theta'_{il}, X_{+I^k_{-l}}(\theta_{il}))$$

$$\Downarrow$$

$$v(\theta'_i, X(\theta_i)) \le \sum_{l=1}^{k} v(\theta'_{il}, X_{+I^k_{-l}}(\theta_{il})).$$

(Here, $X_{+I^k_{-l}}(\theta_{il})$ is short for $X(\theta_{il}, \theta_{-i} \cup (\bigcup_{1\le t\le k, t\neq l} \theta_{it}))$.)

X is said to be *FNP-implementable* if there exists a payment rule P so that X combined with P constitutes a feasible FNP mechanism. Todo *et al.* [102] showed that X is FNP-implementable if and only X satisfies both weak-monotonicity and sub-additivity.

We define allocation rule X to be *FNPW-implementable* if there exists a payment rule P so that X combined with P constitutes a feasible FNPW mechanism. We introduce a third condition called *withdrawal-monotonicity*. We prove that X is FNPW-implementable if and only X satisfies weak-monotonicity, sub-additivity, and withdrawal-monotonicity.

Definition 22. Withdrawal-monotonicity. $\forall\theta_i$, $\forall\theta_{-i}$, $\forall\theta^a$, $\forall\theta^L_i$, $\forall\theta^U_i$, the following holds:

$$v(\theta^L_i, X(\theta^L_i, \theta_{-i})) = 0$$

$$X(\theta^U_i, \theta_{-i} \cup \theta^a) = X(\theta_i, \theta_{-i})$$

$$\Downarrow$$

$$v(\theta^L_i, X(\theta_i, \theta_{-i})) \le v(\theta^U_i, X(\theta_i, \theta_{-i}))$$

Theorem 17. *An allocation rule X is FNPW-implementable if and only X satisfies weak-monotonicity, sub-additivity, and withdrawal-monotonicity.*

Proof. We first prove that if X is FNPW-implementable, then X satisfies weak-monotonicity, sub-additivity, and withdrawal-monotonicity. If X is FNPW-implementable, then X is also FNP-implementable. Hence, X satisfies both weak-monotonicity and sub-additivity [102]; only withdrawal-monotonicity remains to be shown. Let χ be the (PORF) price function corresponding to an FNPW mechanism that allocates according to X. We denote $X(\theta_i, \theta_{-i})$ by S. Since $v(\theta_i^L, X(\theta_i^L, \theta_{-i})) = 0$, we have $v(\theta_i^L, S) \leq \chi(S, \theta_{-i})$ (otherwise, an agent with true type θ_i^L would be better off purchasing S). Since $X(\theta_i^U, \theta_{-i} \cup \theta^a) = X(\theta_i, \theta_{-i}) = S$, we have $v(\theta_i^U, S) \geq \chi(S, \theta_{-i} \cup \theta^a)$ (because an agent with true type θ_i^U is best off buying S when the other agents' types are $\theta_{-i} \cup \theta^a$). χ is FNPW, hence it satisfies the PIA condition, by Theorem 16 and Proposition 69. So, we have $\chi(S, \theta_{-i} \cup \theta^a) \geq \chi(S, \theta_{-i})$. Combining all the inequalities, we get $v(\theta_i^U, X(\theta_i, \theta_{-i})) \geq v(\theta_i^L, X(\theta_i, \theta_{-i}))$. That is, withdrawal-monotonicity is satisfied.

Next, we prove that if X satisfies weak-monotonicity, sub-additivity, and withdrawal-monotonicity, then X is FNPW-implementable. Since X satisfies both weak-monotonicity and sub-additivity, X is FNP-implementable [102]. Let χ be a (PORF) price function that characterizes an FNP mechanism that allocates according to X. We prove that χ must also be FNPW. We only need to prove that χ satisfies PIA (because, according to Proposition 69 and Theorem 16, if an FNP mechanism satisfies PIA, then it is FNPW). Suppose χ does not satisfy PIA. Then, there exists a set of agents O, an agent a not in O (where a's type is denoted by θ^a), and some $S \subseteq G$, such that $\chi(S, O) > \chi(S, O \cup \{a\})$. Let $\chi(S, O) - \chi(S, O \cup \{a\}) = \beta > 0$. Let θ_{-i} be the reported types of the agents in O. Let i be an agent that is single-minded on S, with a very large valuation, so that $X(\theta_i, \theta_{-i}) = S$ (we denote agent i's type by θ_i). We also construct an agent that is single-minded on S, with valuation $\chi(S, O) - \frac{\beta}{3}$. We denote the type of this agent by θ_i^L. We have $X(\theta_i^L, \theta_{-i}) = \emptyset$ (she is not willing to pay $\chi(S, O)$ to purchase S). Hence, $v(\theta_i^L, X(\theta_i^L, \theta_{-i})) = 0$. We construct another agent that is also single-minded on S, with valuation $\chi(S, O \cup \{a\}) + \frac{\beta}{3}$. We denote the type of this agent by θ_i^U. We have $X(\theta_i^U, \theta_{-i} \cup \theta^a) = S = X(\theta_i, \theta_{-i})$. By withdrawal-monotonicity, we must have $v(\theta_i^L, X(\theta_i, \theta_{-i})) \leq v(\theta_i^U, X(\theta_i, \theta_{-i}))$. However, on the other hand, $v(\theta_i^L, X(\theta_i, \theta_{-i})) = \chi(S, O) - \frac{\beta}{3} = \chi(S, O \cup \{a\}) + \frac{2\beta}{3} > \chi(S, O \cup \{a\}) + \frac{\beta}{3} = v(\theta_i^U, X(\theta_i, \theta_{-i}))$. We have reached a contradiction. We conclude that χ has to satisfy PIA, which implies that χ is FNPW. Hence, X is FNPW-implementable. □

4.4 Maximum Marginal Value Item Pricing Mechanism

In this section, we introduce a new FNPW mechanism.

Definition 23. Maximum marginal value item pricing mechanism (MMVIP). Let O be an arbitrary set of agents. MMVIP is characterized by the following price function χ.

- $\forall S \subseteq G$, $\chi(S, O) = \sum_{s \in S} \chi(\{s\}, O)$. That is, χ uses *item pricing.*
- $\forall s \in G$, $\chi(s, O) = \max_{j \in O} \max_{S \subseteq G - \{s\}} \{v(j, S \cup \{s\}) - v(j, S)\}$.[7] That is, the price an agent faces for an item is the maximum possible marginal value that any other agent could have for that item, where the maximum is taken over all possible allocations.

Proposition 72. *MMVIP is feasible and FNPW.*

[7] In this notation, we assume that the maximum over an empty set is 0 (for the purpose of presentation). Such notation will also appear later in the chapter.

Proof. We first prove that MMVIP is feasible. We need to show that, with appropriate tie-breaking, MMVIP will never allocate the same item to multiple agents. Let us suppose that under MMVIP there is a scenario in which two agents, i and j, both win item s. Let S_i and S_j be the sets of other items (items other than s) that i and j win at the end, respectively. Let $v_i = v(i, S_i \cup \{s\}) - v(i, S_i)$. That is, v_i is i's marginal value for s. Let $v_j = v(j, S_j \cup \{s\}) - v(j, S_j)$. That is, v_j is j's marginal value for s. If $v_i > v_j$, then j has to pay at least v_i to win s, which is too high for her; j is better off not winning s. Similarly, if $v_i < v_j$, then i is better off not winning s. If $v_i = v_j$, then i and j both have to pay at least their marginal value for s to win s. That is, they are either indifferent between winning s or not, or prefer not to win. The only case that does not lead to a contradiction is where they are both indifferent; any tie-breaking rule can resolve this conflict.

We then show that MMVIP is FNPW. By Proposition 70, we only need to prove that the price function χ that characterizes MMVIP satisfies S-NSAW. Let O be an arbitrary set of agents. $\forall S_1, S_2 \subseteq G$ with $S_1 \cap S_2 = \emptyset$, we have $\chi(S_1, O) + \chi(S_2, O) = \chi(S_1 \cup S_2, O)$, because MMVIP uses item pricing. Hence, DLB is satisfied. $\forall S \subseteq G$, for any agent a that is not in O, $\chi(S, O \cup \{a\}) = \sum_{s \in S} \chi(s, O \cup \{a\}) = \sum_{s \in S} \max_{j \in O \cup \{a\}} \max_{S' \subseteq G - \{s\}} \{v(j, S' \cup \{s\}) - v(j, S')\} \geq \sum_{s \in S} \max_{j \in O} \max_{S' \subseteq G - \{s\}} \{v(j, S' \cup \{s\}) - v(j, S')\} = \sum_{s \in S} \chi(s, O) = \chi(S, O)$. That is, PIA is also satisfied. □

Next, we prove two properties of the MMVIP mechanism.

Proposition 73. *Suppose we restrict the domain to additive valuations. Then, MMVIP coincides with the VCG mechanism.*

Proof. When the agents' valuations are additive, we have that MMVIP's item price function satisfies $\chi(s, O) = \max_{j \in O} \max_{S \subseteq G - \{s\}} \{v(j, S \cup \{s\}) - v(j, S)\} = \max_{j \in O} v(j, \{s\})$.

Table 4.1: Performance comparison between MMVIP and Set for additive valuations.

	MMVIP	Set
Revenue	98.02	56.19
Efficiency	99.01	57.22

Thus, MMVIP is equivalent to m separate Vickrey auctions (one Vickrey auction for each item), and hence to VCG (which also corresponds to m separate Vickrey auctions when the valuations are additive).

□

The above proposition essentially says that, when the agents' valuations are additive, MMVIP is efficient. MMVIP is the only known FNP/FNPW mechanism with the above property for general combinatorial auctions.

Before moving on to the other property that we prove about MMVIP, we first experimentally compare the revenue and allocative efficiency of the MMVIP mechanism and the Set mechanism, under the assumption that the agents' valuations are additive.[8] We assume that there are 100 items and 100 agents. An agent's valuation for an item is drawn i.i.d. from $U(0, 1)$ (the uniform distribution from 0 to 1). The results are presented in Table 4.1 (the numbers shown are averages over 10000 instances). The experimental results show that MMVIP leads to both higher efficiency and higher revenue.

Finally, we have the following proposition about MMVIP.

Proposition 74. *Among all FNPW mechanisms that use item pricing, MMVIP has minimal payments. That is, let χ be the price function of MMVIP. Let χ' be a different price function corresponding to a different FNPW mechanism M that also*

[8] Under this assumption, the VCG mechanism coincides with the MMVIP mechanism. We also have that the MB mechanism and the Set mechanism coincide. (In our experimental setup, the grand bundle is always a minimal bundle for every agent.)

uses item pricing. We have that there always exists a set of items S and a set of agents O, so that $\chi'(S,O) > \chi(S,O)$.

Proof. For the sake of contradiction, let us assume that the proposition is false. That is, we assume that for every set of items S and every set of agents O, we have $\chi'(S,O) \leq \chi(S,O)$. Since $\chi \neq \chi'$, we have that there exists at least one set of items S and one set of agents O such that $\chi'(S,O) < \chi(S,O)$. Since $\chi'(S,O) = \sum_{s\in S} \chi'(s,O)$ and $\chi(S,O) = \sum_{s\in S} \chi(s,O)$, it follows that there exists $s \in S$ such that $\chi'(s,O) < \chi(s,O)$. By the definition of MMVIP, $\chi(s,O)$ corresponds to the maximal marginal value of some agent $j \in O$. That is, there exists $S' \subset G$ with $s \notin S'$ such that $\chi(s,O) = v(j, S' \cup \{s\}) - v(j, S')$. We construct an agent x, whose valuation function is additive. Let x's valuations of items not in $S' \cup \{s\}$ be extremely high, so that x wins all these items under both mechanisms χ and χ'. (We recall that we assume consumer sovereignty for FNPW mechanisms, so that $\chi, \chi' < \infty$ everywhere.) Let x's valuation on s be $\chi(s,O) - \epsilon$ (where ϵ is small enough so that $\chi(s,O) - \epsilon > \chi'(s,O)$). Let x's valuation of items in S' be 0. When the set of agents consists of x and the agents in O, we have that x wins all the items except for those in S' under M. Since M is FNPW, we have $\chi'(s,O) \geq \chi'(s,\{j\})$ (PIA). That is, when the set of agents consists of only x and j, x also wins all the items except for those in S' under M. Also, under M, j wins all of S', because for any $s' \in S'$, we have $\chi'(s',\{x\}) \leq \chi(s',\{x\}) = 0$. However, we then have that $\chi'(s,\{x\}) \leq \chi(s,\{x\}) = \chi(s,O) - \epsilon = v(j, S' \cup \{s\}) - v(j,S') - \epsilon$, so that j would choose to also win s when facing x under M. That is, under M, when the set of agents consists of only x and j, s is won by both agents, contradicting the assumption that M is feasible. Thus, assuming that the proposition is false leads to a contradiction. □

4.5 Automated FNPW Mechanism Design

In this section, we propose an automated mechanism design (AMD) technique that transforms any feasible mechanism into an FNPW mechanism. In our setting, a feasible mechanism is characterized by a price function χ. We start with any χ that corresponds to a feasible mechanism (*e.g.*, the price function of the VCG mechanism). Our technique modifies χ so that it satisfies S-NSAW, while maintaining feasibility.

We recall that for general combinatorial auction settings, there are three known FNPW mechanisms (Set, MB, and MMVIP), and four known FNP mechanisms (the aforementioned three mechanisms, plus LDS). Though computationally expensive (like many other AMD techniques in other contexts), this technique has the potential to enlarge the set of known FNPW (FNP) mechanisms. By designing tiny instances of FNPW mechanisms via automated mechanism design, we may get a better understanding of the structure of FNPW mechanisms, from which we can then conjecture FNPW mechanisms in analytical form. Later in this section, we show that in a specific setting, by starting with the VCG mechanism, the AMD technique produces exactly the MMVIP mechanism. That is, had we not known the MMVIP mechanism, the AMD technique could have helped us find it (though it just so happened that we discovered MMVIP before the AMD technique). It remains an open question whether new, general FNPW mechanisms can be found in this way.

The AMD technique is described as follows:

Let $H : \Theta^k \rightarrow [0, \infty)$ be a function that maps any set of agents O (more precisely, their reported types) to a nonnegative number $H(O)$. For any feasible mechanism χ, we define χ^H as follows:

- For any set of agents O, $\forall \emptyset \subsetneq S \subseteq G$, $\chi^H(S, O) = \chi(S, O) + H(O)$.

- For any set of agents O, $\chi^H(\emptyset, O) = \chi(\emptyset, O) = 0$.

That is, moving from χ to χ^H, if we fix the reported types of the other agents O, then we are essentially increasing the price of every nonempty set of items by the same amount, while keeping the price of $\emptyset$ at 0.

Lemma 9. *[109] $\forall$ feasible χ, $\forall H$, χ^H is feasible.*

This lemma was first proved in [109].[9] An agent is allocated her favorite set of items (the set that maximizes valuation minus payment) in (PORF) mechanism χ. From the perspective of agent i, the set of types reported by the other agents θ_{-i} is fixed. That is, for i, under χ^H, the price of every nonempty set of items is increased by the same amount $H(\theta_{-i})$. Hence, agent i's favorite set of items is either unchanged, or has become $\emptyset$ (if $H(\theta_{-i})$ is too large). It is thus easy to see that if χ never allocates the same item to more than one agent, then neither does χ^H. That is, feasibility is not affected.[10]

Theorem 18. *$\forall$ feasible χ, we define the following H. For any set of agents O, $H(O)$ equals the maximum of the following two values:*

- $\max_{S_1, S_2 \subseteq G, S_1 \cap S_2 = \emptyset} \{\chi(S_1 \cup S_2, O) - \chi(S_1, O) - \chi(S_2, O)\}$
- $\max_{\emptyset \subsetneq S \subseteq G, j \in O} \{\chi(S, O - \{j\}) + H(O - \{j\}) - \chi(S, O)\}$

We have that χ^H is FNPW.

[9] The GM-SMA mechanism [109] relies on this property. However, it has been shown that GM-SMA is *not* FNP [102].

[10] If the agents are single-minded, then in a PORF mechanism, as long as the prices of larger sets of items are more expensive, an agent's favorite set of items is either the set on which she is single-minded, or the empty set. Thus, we do not need to increase the price of every set by the same amount. As long as we are increasing the prices, an agent's favorite set either remains unchanged, or becomes empty (if the price increase on the set on which she is single-minded is too high). That is, for single-minded agents, we have more flexibility in the process of transforming a feasible mechanism into an FNPW mechanism.

It should be noted that, for any O, the first expression in the theorem is at least 0 (setting $S_1 = S_2 = \emptyset$). That is, H never takes negative values. χ^H is feasible by Lemma 9.

Proof. We prove that χ^H satisfies S-NSAW. By Proposition 70, this suffices to show that χ^H is FNPW.

Proof of DLB: Let O be an arbitrary set of agents. $\forall S_1, S_2 \subseteq G$ with $S_1 \cap S_2 = \emptyset$, we prove that $\chi^H(S_1, O) + \chi^H(S_2, O) \geq \chi^H(S_1 \cup S_2, O)$. If at least one of S_1 and S_2 is empty, then w.l.o.g., we assume $S_1 = \emptyset$. In this case, $\chi^H(S_1, O) + \chi^H(S_2, O) = \chi^H(S_2, O) = \chi^H(S_1 \cup S_2, O)$. If neither S_1 nor S_2 is empty, then we have $\chi^H(S_1, O) + \chi^H(S_2, O) - \chi^H(S_1 \cup S_2, O) = H(O) + \chi(S_1, O) + \chi(S_2, O) - \chi(S_1 \cup S_2, O) \geq H(O) - \max_{S_1' \cap S_2' = \emptyset} \{\chi(S_1' \cup S_2', O) - \chi(S_1', O) - \chi(S_2', O)\} \geq 0$.

Proof of PIA: Let O be an arbitrary set of agents. Let a be an agent that is not in O. If S is empty, then we have $\chi^H(S, O \cup \{a\}) = \chi^H(S, O) = 0$. $\forall \emptyset \subsetneq S \subseteq G$, $\chi^H(S, O \cup \{a\}) = H(O \cup \{a\}) + \chi(S, O \cup \{a\}) \geq (\chi(S, O) + H(S, O) - \chi(S, O \cup \{a\})) + \chi(S, O \cup \{a\}) = \chi^H(S, O)$. ∎

This still leaves the question of how to compute the H described in the theorem; we address this next. Given χ, for any agent i and any set of other types θ_{-i}, we compute $H(\theta_{-i})$ using the following dynamic programming algorithm.

For $t = 0, 1, \ldots, |\theta_{-i}|$

  For any $T \subseteq \theta_{-i}$ with $|T| = t$

    $h_1 = \max_{S_1, S_2 \subseteq G, S_1 \cap S_2 = \emptyset} \{\chi(S_1 \cup S_2, T) - \chi(S_1, T) - \chi(S_2, T)\}$.
    $h_2 = \max_{\emptyset \subsetneq S \subseteq G, j \in T} \{H(T - \{j\}) + \chi(S, T - \{j\}) - \chi(S, T)\}$.
    $H(T) = \max\{h_1, h_2\}$.

It should be noted that the above dynamic programming algorithm does not scale well. For example, the second "For" loop contains an exponential number of steps.

Proposition 75. *If we apply the AMD technique to a mechanism that already satisfies S-NSAW, the mechanism remains unchanged.*

We use the phrase "the AMD mechanism" to denote the mechanism generated by the AMD technique starting from VCG (though the AMD technique is not restricted to starting from VCG). Next, we prove a proposition that is similar to Proposition 73.

Proposition 76. *When we restrict the preference domain to additive valuations, the MMVIP, VCG, and AMD mechanism all coincide.*

Proof. Proposition 73 already shows that MMVIP and VCG coincide. All that remains to show is that VCG already satisfies S-NSAW, so that by Proposition 75, AMD is also the same. When the agents' valuations are additive, the VCG mechanism's price function χ is defined as follows: for any set of items $S \subset G$ and any set of additive agents O, $\chi(S, O) = \sum_{s \in S} x^s$, where x^s is the highest valuation for item s among the agents in O. It is easy to see that χ satisfies S-NSAW. □

Moreover, the next proposition shows that in settings with exactly two substitutable items, the AMD mechanism coincides with MMVIP (but not with VCG).

Proposition 77. *In settings with exactly two substitutable items, the AMD mechanism coincides with MMVIP.*

Proof. The proof is by induction on the number of agents. When there is only one agent, this agent faces price 0 for every bundle under the VCG mechanism. This already satisfies S-NSAW, so by Proposition 75, we do not need to increase any price in the AMD process. Therefore, when $n = 1$, the AMD mechanism allocates all the items to the only agent for free. The MMVIP mechanism does the same. Hence, when $n = 1$, the AMD mechanism coincides with MMVIP. For the induction step, we assume that the two mechanisms coincide when $n \leq k$. When $n = k+1$, the price function of the VCG mechanism is defined as: $\chi(\{A\}, O) = v^*_{AB} - v^*_B$, $\chi(\{B\}, O) = v^*_{AB} - v^*_A$, and $\chi(\{AB\}, O) = v^*_{AB}$. Here, A and B are the two items. v^*_A is the highest valuation for A by the agents in O. v^*_B is the highest valuation for B by the agents in O. v^*_{AB} is the highest combined valuation for $\{A, B\}$ by the agents in O (which may be obtained by splitting the items across two different agents, or giving both to the same agent). Since the items are substitutable, $v^*_{AB} \leq v^*_A + v^*_B$. Equivalently, $\chi(\{A\}, O) + \chi(\{B\}, O) \leq \chi(\{AB\}, O)$. Therefore, in the AMD technique, the price of every bundle has to increase by at least $\chi(\{A, B\}, 0) - \chi(\{A\}, O) - \chi(\{B\}, O)$. That is, under the AMD mechanism, the price of A is at least v^*_A, the price of B is at least v^*_B, and the price of $\{A, B\}$ is at least $v^*_A + v^*_B$. These prices are high enough to guarantee the PIA condition, because by the induction assumption, the AMD mechanism coincides with MMVIP for $n \leq k$; so, it follows that the AMD technique results in exactly these prices. They coincide with the prices under the MMVIP mechanism. Therefore, by induction, the AMD mechanism coincides with the MMVIP mechanism for any number of agents, when there are exactly two substitutable items. □

It remains an open question whether there are more general settings in which the AMD mechanism and the MMVIP mechanism coincide.

Finally, we compare the revenue and allocative efficiency of the VCG mechanism, the Set mechanism[11], the MMVIP mechanism, and the AMD mechanism. It should be noted that the VCG mechanism is not FNPW in general. We use it as a benchmark.

We consider a combinatorial auction with two items $\{A, B\}$ and five agents $\{1, 2, \ldots, 5\}$.[12] We denote agent i's valuation for set $S \subseteq \{A, B\}$ by v_i^S. We consider

[11] The MB mechanism and the Set mechanism coincide in our experimental setup (the whole bundle is a minimal bundle for every agent).

[12] We only focused on these tiny auctions because the AMD technique is computationally quite

Table 4.2: Performance comparison between VCG, Set, AMD, and MMVIP for substitutable valuations.

	VCG	Set	AMD	MMVIP
Revenue	1.285	1.002	1.221	1.221
Efficiency	1.668	1.236	1.550	1.550

Table 4.3: Performance comparison between VCG, Set, AMD, and MMVIP for complementary valuations.

	VCG	Set	AMD	MMVIP
Revenue	1.864	1.849	1.288	0.594
Efficiency	2.372	2.365	1.565	0.721

two scenarios, one with valuations displaying substitutability, and the other with valuations displaying complementarity. We randomly generate 1000 instances for each scenario.

Valuations with substitutability: The $v_i^{\{A\}}$ and the $v_i^{\{B\}}$ are drawn independently from $U(0,1)$ (the uniform distribution from 0 to 1). For all i, $v_i^{\{A,B\}}$ is drawn independently from $U(\max\{v_i^{\{A\}}, v_i^{\{B\}}\}, v_i^{\{A\}} + v_i^{\{B\}})$. In this scenario, AMD and MMVIP coincide. They perform better than the Set mechanism, both in terms of revenue and allocative efficiency. The results are presented in Table 4.2.

Valuations with complementarity: The $v_i^{\{A\}}$ and the $v_i^{\{B\}}$ are still drawn independently from $U(0,1)$. For all i, $v_i^{\{A,B\}}$ is set to be $(v_i^{\{A\}} + v_i^{\{B\}})(1 + x_i)$, where the x_i are also drawn independently from $U(0,1)$. It turns out that, in this scenario, Set performs better than AMD and MMVIP, both in terms of revenue and allocative efficiency. (MMVIP performs especially poorly when valuations exhibit complementarity, because every item can potentially have a very large marginal value to another agent, leading to prices that are too high.) The results are presented in Table 4.3.

Thus, when there are two items and five agents, among these FNPW mechanisms,

expensive. Nevertheless, even the solutions to tiny auctions can be helpful in conjecturing more general mechanisms.

it seems that Set is most desirable if it likely that there is significant complementarity, and AMD is most desirable if it is likely that there is substitutability. (We cannot safely use the VCG mechanism unless we are certain that the type space makes VCG FNPW.)

4.6 Worst-Case Efficiency Ratio of FNPW Mechanisms

Yokoo *et al.* [111] proved that in general combinatorial auction settings, there exists no efficient FNP mechanisms. Iwasaki *et al.* [69] further showed that, under a minor condition called IIG (described below), the worst-case efficiency ratio of any feasible FNP mechanism is at most $\frac{2}{m+1}$.[13]

Definition 24. Independence of irrelevant good (IIG) [69]. Suppose agent i is winning all the items. If we add an additional item that is only wanted by i, then i still wins all the items.

Given the agents' reported types, the efficiency ratio of a mechanism is defined as the ratio between the achieved allocative efficiency and the optimal allocative efficiency (payments are not taken into consideration). The worst-case efficiency ratio of this mechanism is the minimal such ratio over all possible type profiles.

Example 19. *The worst-case efficiency ratio of the Set mechanism is at least $\frac{1}{m}$ [69].* Let v be the winning agent's valuation for the grand bundle. The allocative efficiency of the Set mechanism is v. The optimal allocative efficiency is at most mv, since there are at most m winners in the optimal allocation, and a winner's valuation (for the items she won) is at most v.

Our next theorem is that $\frac{1}{m}$ is a strict upper bound on the efficiency ratios of feasible FNPW mechanisms. That is, the Set mechanism is worst-case optimal in

[13] Iwasaki *et al.* [69] also introduced the ARP mechanism, whose worst-case efficiency ratio is exactly $\frac{2}{m+1}$. However, the ARP mechanism is only FNP for single-minded agents. Our next result implies that ARP is not FNPW, even with single-minded bidders.

terms of efficiency ratio. Of course, this is only a worst-case analysis, which does not preclude FNPW mechanisms from performing well most of the time.

Theorem 19. *The worst-case efficiency ratio of any feasible FNPW mechanism is at most* $\frac{1}{m}$ *if IIG holds, even with single-minded bidders.*

Proof. Let χ be the price function that corresponds to an FNPW mechanism with optimal worst-case ratio. Since the Set mechanism is FNPW, χ's worst-case efficiency ratio is at least $\frac{1}{m}$. We denote item i by s_i. We consider the following types:

θ_a: the type of an agent that is single-minded on the grand bundle, with value 1.

θ_i ($i = 1, 2, \ldots, m$): the type of an agent that is single-minded on s_i, with value $1 - \epsilon$. Here, ϵ is a small positive number.

Scenario 1: There are two agents. Agent a has type θ_a. Agent 1 has type θ_1.

Scenario 2: There are two agents. Both agents have type θ_1.

Scenario 3: There are $m+1$ agents. Agent a has type θ_a. Agent i has type θ_i for $i = 1, 2, \ldots, m$.

We first prove that in scenario 1, agent a wins. We start with the special case of $m = 1$. If $\chi(\{s_1\}, \{\theta_1\}) > 1 - \epsilon$, then we consider scenario 2. In scenario 2, both agents cannot afford the only item. That is, the efficiency ratio is 0. Hence, we must have $\chi(\{s_1\}, \{\theta_1\}) \leq 1 - \epsilon$. That is, in scenario 1, in the case of $m = 1$, agent a must win. The IIG condition implies that this is also true for cases with $m > 1$.

Since agent a is the only winner in scenario 1, we have $\chi(\{s_1\}, \{\theta_a\}) \geq 1 - \epsilon$ (otherwise, agent 1 would win in scenario 1). ϵ can be made arbitrarily close to 0; hence, $\chi(\{s_1\}, \{\theta_a\}) \geq 1$.

Finally, we consider scenario 3. The price agent 1 faces for s_1 is $\chi(\{s_1\}, \{\theta_a\} \cup (\bigcup_{j \neq 1} \{\theta_j\}))$. According to PIA, this price is at least $\chi(\{s_1\}, \{\theta_a\}) = 1$. That is, agent 1 does not win in scenario 3. By symmetry over the items, agent i does not win for all $i = 1, 2, \ldots, m$. The efficiency ratio in this scenario is then at most $\frac{1}{m(1-\epsilon)}$, which goes to $\frac{1}{m}$ as ϵ goes to 0. □

4.7 Characterizing FNP(W) in Social Choice Settings

Throughout the chapter, we have only discussed combinatorial auctions. In this section, we focus on FNP(W)[14] in social choice settings (without payments). Specifically, we present a characterization of FNP(W) social choice functions (without payments). A social choice function f is defined as $f : \{\emptyset\} \cup \Theta \cup \Theta^2 \cup \ldots \to \Omega$, where Θ is the space of all possible types of an agent, and $\{\emptyset\} \cup \Theta \cup \Theta^2 \cup \ldots$ is the space of all possible profiles (since we do not know how many agents there are). The definitions of Ω, θ_i, and θ_{-i} are as usual. i's valuation for outcome $\omega \in \Omega$ is denoted by $v_i(\theta_i, \omega)$.

First, we present the following straightforward characterization of strategy-proof social choice functions.

Proposition 78. *A social choice function f is strategy-proof if and only if it satisfies the following condition: $\forall i, \theta_i, \theta_{-i}$, we have $f(\theta_i, \theta_{-i}) \in \arg\max_{\theta'_i} v_i(\theta_i, f(\theta'_i, \theta_{-i}))$.*

Proof. If the above condition is satisfied, then $\forall i, \theta_i, \theta'_i, \theta_{-i}$, we have $v_i(\theta_i, f(\theta_i, \theta_{-i})) \geq v_i(\theta_i, f(\theta'_i, \theta_{-i}))$. That is, reporting truthfully is a dominant strategy.

If reporting truthfully is a dominant strategy, then $\forall i, \theta_i, \theta'_i, \theta_{-i}$, we have $v_i(\theta_i, f(\theta_i, \theta_{-i})) \geq v_i(\theta_i, f(\theta'_i, \theta_{-i}))$. That is, $\forall i, \theta_i, \theta_{-i}$, we have $v_i(\theta_i, f(\theta_i, \theta_{-i})) \geq \max_{\theta'_i} v_i(\theta_i, f(\theta'_i, \theta_{-i}))$, which is equivalent to $f(\theta_i, \theta_{-i}) \in \arg\max_{\theta'_i} v_i(\theta_i, f(\theta'_i, \theta_{-i}))$. □

That is, an agent always receives her most-preferred choice among outcomes that

[14] In these settings, it does not matter whether withdrawal is allowed or not.

she can attain with some type report. We are now ready to present the characterization of FNP(W) social choice functions.

Proposition 79. *Suppose that for every outcome $o \in \Omega$, there exists some type $\theta_i \in \Theta$ such that $\{o\} = \arg\max_{o' \in O} u_{\theta_i}(o')$ (each o is the unique most-preferred outcome for some type). Then, a strategy-proof and individually rational social choice function f is FNP(W) if and only if it satisfies the following condition: $\forall i, \theta_{-i}, \theta_0$, we have $\{f(\theta_i, \theta_{-i}) | \theta_i \in \Theta\} \supseteq \{f(\theta_i, \theta_{-i} \cup \{\theta_0\}) | \theta_i \in \Theta\}$. That is, with an additional other agent, the set of outcomes that an agent can choose decreases or stays the same.*

Proof. We first show that if f is FNP(W), then the condition must be satisfied. Suppose not, that is, for some i, θ_{-i}, θ_0, there exists some $o \in \{f(\theta_i, \theta_{-i} \cup \{\theta_0\}) | \theta_i \in \Theta\} \setminus \{f(\theta_i, \theta_{-i}) | \theta_i \in \Theta\}$. Then, by assumption, there exists some $\theta_i \in \Theta$ such that $\{o\} = \arg\max_{o' \in O} u_{\theta_i}(o')$. It follows that an agent facing type profile θ_{-i} cannot obtain o with a single report, but can obtain it by reporting both θ_0 and some other type (such as, by strategy-proofness, θ_i). Because o is her unique most-preferred outcome, she prefers to engage in this manipulation, contradicting FNP(W).

Conversely, we show that if the condition is satisfied, then f is FNP(W). By assumption, f is strategy-proof and individually rational, so we only need to check that an agent has no incentive to use multiple identifiers. Suppose that o is an outcome that i can obtain when facing θ_{-i} by submitting multiple identities. Because the set of choices is nonincreasing in the number of identifiers used according to the condition, it must be that $o \in \{f(\theta_i, \theta_{-i}) | \theta_i \in \Theta\}$. Hence, there is no reason for her to use more than one identity. □

The above proposition basically says that under FNP(W) social choice functions, with the introduction of a new agent, an existing agent's set of possible choices decreases (it potentially hurts the existing agent's utility) or stays the same (essentially, it is equivalent to ignoring the new agent's preference). That is, with a large number of agents, generally, FNP(W) social choice functions perform poorly.

4.8 Summary

In this chapter, we studied a more powerful variant of false-name manipulation: an agent can submit multiple false-name bids, but then, once the allocation and payments have been decided, withdraw some of her false-name identities. While these withdrawn identities will not obtain the items they won, their initial presence may have been beneficial to the agent's other identities. A mechanism is *false-name-proof with withdrawal (FNPW)* if the aforementioned manipulation is never beneficial under it. We first gave a necessary and sufficient condition on the type space for the VCG mechanism to be FNPW. We then characterized both the payment rules and the allocation rules of FNPW mechanisms in general combinatorial auctions. Based on the characterization of the payment rules, we derived a condition that is sufficient for a mechanism to be FNPW. We also proposed the *maximum marginal value item pricing (MMVIP)* mechanism. We showed that MMVIP is FNPW and exhibit some of its desirable properties. We then proposed an automated mechanism design technique that transforms any feasible mechanism into an FNPW mechanism, and proved some basic properties about this technique. Toward the end, we proved a strict upper bound on the worst-case efficiency ratio of FNPW mechanisms. We concluded with a characterization of FNP(W) social choice rules. Since FNPW is stronger than FNP, this chapter also contributes to the research on false-name-proofness in the traditional sense.

5

Conclusion

In this dissertation, we formalized an approach to automated mechanism design that is computationally feasible. Instead of optimizing over all feasible mechanisms, we carefully choose a parameterized subfamily of mechanisms. Then we optimize over mechanisms within this family. Finally, we analyze whether and to what extent the resulting mechanism is suboptimal outside the subfamily. We applied (computationally feasible) automated mechanism design to three resource allocation mechanism design problems:

- In Chapter 2, we applied CFAMD to the problem of designing resource allocation mechanisms that redistribute their revenue back to the agents.

  - In Section 2.1, we focused on designing VCG redistribution mechanisms that redistribute the most in the *worst case*. For auctions with multiple indistinguishable units in which marginal values are nonincreasing, we derived a mechanism that is optimal in this sense. We also showed that if marginal values are not required to be nonincreasing, then the original VCG mechanism is worst-case optimal. For future research, we could consider designing worst-case optimal redistribution mechanisms for other problem settings. Example work in this direction includes the following: Gujar and Yadati [53] conjectured that for auctions with multiple heterogeneous objects in which each agent is only interested in one object, the worst-case optimal mechanism has the same worst-case performance as the WCO mechanism. Chorppath *et al.* [24] studied worst-case redistribution for auctions with divisible goods.

– In Section 2.2, we studied the problem of designing VCG redistribution mechanisms that redistribute the most in *expectation* when prior distributions over the agents' valuations are available. For auctions with multiple indistinguishable units in which each agent is only interested in one unit, we analytically derived the OEL mechanism that is optimal among linear redistribution mechanisms. For this setting, we also proposed an automated mechanism design technique based on type discretization. We then generalized our setting to auctions with multiple indistinguishable units in which marginal values are nonincreasing. We extended the notion of linear redistribution mechanisms to this more general setting. In the more general setting, optimization within the family of linear redistribution mechanisms becomes more difficult, because we need to consider a type of ordering information. If we completely ignore the ordering information, then the resulting mechanisms generally do not perform well. It remains to see whether we can identify a subfamily of linear redistribution mechanisms that is easy to optimize over, and still captures some of the ordering information.

– In Section 2.3, we studied the problem of designing mechanisms whose redistribution functions are *undominated* in the sense that no other mechanisms can always perform as well, and sometimes better. We introduced two measures (individual and collective dominance) for comparing two VCG redistribution mechanisms with respect to how well off they make the agents, and studied the question of finding maximal elements in the space of non-deficit redistribution mechanisms, with respect to the partial orders induced by both measures. Most of our positive results only apply to auctions with multiple indistinguishable units, where each agent is only interested in a single copy of the unit. For example, in this setting, we characterized the (individually and collectively) undominated redistribution mechanisms that are linear and anonymous. It remains to see whether we can characterize more undominated mechanisms in other settings. In this section, we also gave two techniques for transforming existing individually dominated mechanisms into mechanisms that are individually undominated. It remains to see whether we can derive similar techniques for generating collectively undominated mechanisms.

  - In Section 2.4, we studied the problem of designing the allocation rule together with the redistribution scheme, allowing for the allocation to be inefficient. We proposed several specific mechanisms that are based on burning items, excluding agents, and (most generally) partitioning the items and agents into groups. The mechanisms we proposed are not guaranteed to be optimal. de Clippel *et al.* [42] studied deterministic mechanisms that are based on burning items in more detail, and obtained mechanisms that achieve higher competitive ratios. It is still an open question to derive optimal mechanisms that are based on inefficient allocation.

- In Chapter 3, we applied CFAMD to the problem of designing resource allocation mechanisms that do not rely on payments at all.

- In Section 3.1, we studied the problem of allocating a single item repeatedly among multiple competing agents, in an environment where monetary transfers are not possible. We introduced an artificial payment system, which enabled us to construct repeated allocation mechanisms without payments based on one-shot allocation mechanisms with payments. Under certain restrictions on the discount factor, we proposed several (Bayes-Nash) incentive compatible repeated allocation mechanisms based on artificial payments. We proved that our mechanisms are competitive against the first-best allocation. The artificial payment system we proposed is based on a triangular approximation of the optimal frontier. It would be interesting to study artificial payment systems based on other forms of approximation. With other artificial payment systems, we may be able to construct mechanisms with higher competitive ratios.

- In Section 3.2, we investigated the problem of allocating multiple items among two competing agents in a (single-round) setting that is both prior-free and payment-free. We introduced the family of linear increasing-price (LIP) mechanisms. The LIP mechanisms are strategy-proof and only rely on artificial payments. We showed how to solve for mechanisms within the LIP family that are competitive against the first-best allocation. For very small numbers of items, we are able to find LIP mechanisms that perform well. However, as the number of items increases, the competitive ratio of the optimal LIP mechanism goes to $\frac{1}{2}$ (the competitive ratio of the naïve lottery mechanism). It is still an open problem to design mechanisms with high competitive ratios for large numbers of items.

• In Chapter 4, we studied a more powerful variant of false-name manipulation: an agent can submit multiple false-name bids, but then, once the allocation and payments have been decided, withdraw some of her false-name identities. While these withdrawn identities will not obtain the items they won, their initial presence may have been beneficial to the agent's other identities. A mechanism is *false-name-proof with withdrawal (FNPW)* if the aforementioned manipulation is never beneficial under it. We first gave a necessary and sufficient condition on the type space for the VCG mechanism to be FNPW. We then characterized both the payment rules and the allocation rules of FNPW mechanisms in general combinatorial auctions. Based on the characterization of the payment rules, we derived a condition that is sufficient for a mechanism to be FNPW. We also proposed the *maximum marginal value item pricing (MMVIP)* mechanism. We showed that MMVIP is FNPW and exhibited some of its desirable properties. We then proposed an automated mechanism design technique that transforms any feasible mechanism into an FNPW mechanism, and proved some basic properties about this technique. Toward the end, we proved a strict upper bound on the worst-case efficiency ratio of FNPW mechanisms. We concluded with a characterization of FNP(W) social choice rules. Since FNPW is stronger than FNP, this chapter also contributes to the research on false-name-proofness in the traditional sense. For future research, we could try to improve the AMD technique, so that we are able to design new FNP(W) mechanisms based on AMD.

Finally, on a higher level, we believe that the computationally feasible automated mechanism design approach, where the analytical capabilities of a human mechanism designer work in concert with algorithms that search through restricted families of possible mechanisms, is currently the most promising avenue for techniques from artificial intelligence to contribute to the theory of mechanism design and (perhaps) microeconomic theory in general. The human mechanism designer plays an essential role at several points in this process. For future research on CFAMD, it would be desirable to find ways to reduce the burden that is placed on the human designer. For example, can we design algorithms that automatically conjecture natural families of mechanisms? Can we design algorithms that generalize the solutions to particular instances into a general analytical form? Can we design algorithms that automatically analyze how suboptimal a given mechanism is relative to the space of all feasible mechanisms? These are challenging, but not unimaginable ways in which AI can take on an even greater role in proving results in mechanism design and beyond.

Bibliography

[1] Atila Abdulkadiroğlu and Kyle Bagwell. Trust, reciprocity and favors in cooperative relationships. Working paper, 2005.

[2] Dilip Abreu, David Pearce, and Ennio Stacchetti. Toward a theory of discounted repeated games with imperfect monitoring. *Econometrica*, 58(5):1041–63, September 1990.

[3] Krzysztof Apt, Vincent Conitzer, Mingyu Guo, and Evangelos Markakis. Welfare undominated Groves mechanisms. In *Proceedings of the Fourth Workshop on Internet and Network Economics (WINE)*, pages 426–437, Shanghai, China, 2008.

[4] Susan Athey and Kyle Bagwell. Optimal collusion with private information. Working papers 99-17, Massachusetts Institute of Technology (MIT), Department of Economics, September 1999.

[5] Susan Athey, Kyle Bagwell, and Chris Sanchirico. Collusion and price rigidity. *Review of Economic Studies*, 71(2):317–349, 04 2004.

[6] Susan Athey and Ilya Segal. An efficient dynamic mechanism. Technical report, 2007. Working Paper.

[7] Lawrence M. Ausubel and Paul Milgrom. The lovely but lonely Vickrey auction. In Peter Cramton, Yoav Shoham, and Richard Steinberg, editors, *Combinatorial Auctions*, chapter 1, pages 17–40. MIT Press, 2006.

[8] Martin J. Bailey. The demand revealing process: to distribute the surplus. *Public Choice*, 91:107–126, 1997.

[9] Maria-Florina Balcan, Avrim Blum, Jason D. Hartline, and Yishay Mansour. Mechanism design via machine learning. In *Proceedings of the Annual Symposium on Foundations of Computer Science (FOCS)*, pages 605–614, 2005.

[10] Salvador Barberà. An introduction to strategy-proof social choice functions. *Social Choice and Welfare*, 18(4):619–653, 2001.

[11] Dirk Bergemann and Juuso Välimäki. Efficient dynamic auctions. Cowles Foundation Discussion Papers 1584, Cowles Foundation, Yale University, October 2006.

[12] Sayan Bhattacharya, Gagan Goel, Sreenivas Gollapudi, and Kamesh Munagala. Budget constrained auctions with heterogeneous items. In *STOC '10: Proceedings of the 42nd ACM symposium on Theory of computing*, pages 379–388, Cambridge, Massachusetts, USA, 2010.

[13] Sushil Bikhchandani, Shurojit Chatterji, Ron Lavi, Ahuva Mu'alem, Noam Nisan, and Arunava Sen. Weak monotonicity characterizes deterministic dominant strategy implementation. *Econometrica*, 74(4):1109–1132, 2006.

[14] S. J. Brams and A. D. Taylor. *The Win-Win Solution: Guaranteeing Fair Shares to Everybody.* W. W. Norton & Company, 1999.

[15] Steven J. Brams, Michael A. Jones, and Christian Klamler. Better ways to cut a cake - revisited. In Steven Brams, Kirk Pruhs, and Gerhard Woeginger, editors, *Fair Division*, number 07261 in Dagstuhl Seminar Proceedings, Dagstuhl, Germany, 2007. Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Germany.

[16] Eric Budish. The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes, November 2008. Working Paper.

[17] Eric Budish and Estelle Cantillon. Strategic behavior in multi-unit assignment problems: Theory and evidence from course allocations. In Peter Cramton, Rudolf Müller, Eva Tardos, and Moshe Tennenholtz, editors, *Computational Social Systems and the Internet*, number 07271 in Dagstuhl Seminar Proceedings, Dagstuhl, Germany, 2007. Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Germany.

[18] Andrew Byde. Applying evolutionary game theory to auction mechanism design. In *Proceedings of the ACM Conference on Electronic Commerce (EC)*, pages 192–193, San Diego, CA, USA, 2003. Poster paper.

[19] Alessandra Casella. Storable votes. *Games and Economic Behavior*, 51:391–419, 2005.

[20] Ruggiero Cavallo. Optimal decision-making with minimal waste: Strategyproof redistribution of VCG payments. In *Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 882–889, Hakodate, Japan, 2006.

[21] Ruggiero Cavallo. Efficiency and redistribution in dynamic mechanism design. In *Proceedings of the ACM Conference on Electronic Commerce (EC)*, pages 220–229, Chicago, IL, USA, 2008.

[22] Ruggiero Cavallo, David C. Parkes, and Satinder Singh. Optimal coordinated planning amongst self-interested agents with private state. In *Proceedings of the 22nd Annual Conference on Uncertainty in Artificial Intelligence (UAI)*, Cambridge, MA, USA, 2006.

[23] Yiling Chen, John Lai, David Parkes, and Ariel Procaccia. Truth, justice, and cake cutting. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, Atlanta, GA, USA, 2010.

[24] Anil Kumar Chorppath, Srikrishna Bhashyam, and Rajesh Sundaresan. Almost budget-balanced mechanisms for allocation of divisible resources. In *Allerton'09: Proceedings of the 47th annual Allerton conference on Communication, control, and computing*, pages 1120–1125, Monticello, Illinois, USA, 2009.

[25] Ed H. Clarke. Multipart pricing of public goods. *Public Choice*, 11:17–33, 1971.

[26] Dave Cliff. Evolution of market mechanism through a continuous space of auction-types. Technical Report HPL-2001-326, HP Labs, 2001.

[27] Vincent Conitzer. *Computational aspects of preference aggregation*. PhD thesis, Carnegie Mellon University, 2006.

[28] Vincent Conitzer. Limited verification of identities to induce false-name-proofness. In *Theoretical Aspects of Rationality and Knowledge (TARK)*, pages 102–111, Brussels, Belgium, 2007.

[29] Vincent Conitzer. Anonymity-proof voting rules. In *Proceedings of the Fourth Workshop on Internet and Network Economics (WINE)*, pages 295–306, Shanghai, China, 2008.

[30] Vincent Conitzer and Tuomas Sandholm. Complexity of mechanism design. In *Proceedings of the 18th Annual Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 103–110, Edmonton, Canada, 2002.

[31] Vincent Conitzer and Tuomas Sandholm. Applications of automated mechanism design. In *UAI-03 workshop on Bayesian Modeling Applications*, Acapulco, Mexico, 2003.

[32] Vincent Conitzer and Tuomas Sandholm. Automated mechanism design: Complexity results stemming from the single-agent setting. In *Proceedings of the 5th International Conference on Electronic Commerce (ICEC-03)*, pages 17–24, Pittsburgh, PA, USA, 2003.

[33] Vincent Conitzer and Tuomas Sandholm. An algorithm for automatically designing deterministic mechanisms without payments. In *Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 128–135, New York, NY, USA, 2004.

[34] Vincent Conitzer and Tuomas Sandholm. Self-interested automated mechanism design and implications for optimal combinatorial auctions. In *Proceedings of the ACM Conference on Electronic Commerce (EC)*, pages 132–141, New York, NY, USA, 2004.

[35] Vincent Conitzer and Tuomas Sandholm. Failures of the VCG mechanism in combinatorial auctions and exchanges. In *Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 521–528, Hakodate, Japan, 2006.

[36] Vincent Conitzer and Tuomas Sandholm. Incremental mechanism design. In *Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1251–1256, Hyderabad, India, 2007.

[37] Florin Constantin and David Parkes. On revenue-optimal dynamic auctions for bidders with interdependent values. In *AMEC*, Honolulu, HI, USA, 2007.

[38] Florin Constantin and David C. Parkes. Self-correcting sampling-based dynamic multi-unit auctions. In *Proceedings of the ACM Conference on Electronic Commerce (EC)*, Stanford, CA, USA, 2009.

[39] Peter Cramton, Robert Gibbons, and Paul Klemperer. Dissolving a partnership efficiently. *Econometrica*, 55(3):615–632, 1987.

[40] Peter Cramton, Yoav Shoham, and Richard Steinberg. *Combinatorial Auctions*. MIT Press, 2006.

[41] Claude d'Aspremont and Louis-Andre Gérard-Varet. Incentives and incomplete information. *Journal of Public Economics*, 11:25–45, 1979.

[42] Geoffroy de Clippel, Victor Naroditskiy, and Amy Greenwald. Destroy to save. In *Proceedings of the ACM Conference on Electronic Commerce (EC)*, pages 207–214, Stanford, CA, USA, 2009.

[43] Ofer Dekel, Felix Fischer, and Ariel D. Procaccia. Incentive compatible regression learning. In *Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 884–893, Philadelphia, PA, USA, 2008. Society for Industrial and Applied Mathematics.

[44] Lars Ehlers and Bettina Klaus. Coalitional strategy-proof and resource-monotonic solutions for multiple assignment problems. *Social Choice and Welfare*, 21(2):265–280, October 2003.

[45] Boi Faltings. A budget-balanced, incentive-compatible scheme for social choice. In *Agent-Mediated Electronic Commerce (AMEC), LNAI, 3435*, pages 30–43, 2005.

[46] Joan Feigenbaum, Christos H. Papadimitriou, and Scott Shenker. Sharing the cost of mulicast transmissions. *Journal of Computer and System Sciences*, 63:21–41, 2001.

[47] Eric J. Friedman, Joseph Y. Halpern, and Ian Kash. Efficiency and nash equilibria in a scrip system for p2p networks. In *Proceedings of the ACM Conference on Electronic Commerce (EC)*, pages 140–149, Ann Arbor, MI, USA, 2006.

[48] Drew Fudenberg, David I Levine, and Eric Maskin. The folk theorem with imperfect public information. *Econometrica*, 62(5):997–1039, September 1994.

[49] Allan Gibbard. Manipulation of voting schemes: a general result. *Econometrica*, 41:587–601, 1973.

[50] Andrew Goldberg, Jason Hartline, Anna Karlin, Mike Saks, and Andrew Wright. Competitive auctions. *Games and Economic Behavior*, 2006.

[51] Jerry Green and Jean-Jacques Laffont. Characterization of satisfactory mechanisms for the revelation of preferences for public goods. *Econometrica*, 45:427–438, 1977.

[52] Theodore Groves. Incentives in teams. *Econometrica*, 41:617–631, 1973.

[53] Sujit Gujar and Narahari Yadati. Redistribution of vcg payments in assignment of heterogeneous objects. In *Proceedings of the Fourth Workshop on Internet and Network Economics (WINE)*, pages 438–445, Shanghai, China, 2008.

[54] Mingyu Guo and Vincent Conitzer. Better redistribution with inefficient allocation in multi-unit auctions with unit demand. In *Proceedings of the ACM Conference on Electronic Commerce (EC)*, pages 210–219, Chicago, IL, USA, 2008.

[55] Mingyu Guo and Vincent Conitzer. Undominated VCG redistribution mechanisms. In *Proceedings of the Seventh International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 1039–1046, Estoril, Portugal, 2008.

[56] Mingyu Guo and Vincent Conitzer. Worst-case optimal redistribution of VCG payments in multi-unit auctions. *Games and Economic Behavior*, 67(1):69–98, 2009. Special Section of Games and Economic Behavior Dedicated to the 8th ACM Conference on Electronic Commerce.

[57] Mingyu Guo and Vincent Conitzer. Computationally feasible automated mechanism design: General approach and case studies. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, Atlanta, GA, USA, 2010. NECTAR track.

[58] Mingyu Guo and Vincent Conitzer. False-name-proofness with bid withdrawal (extended abstract). In *Proceedings of the Ninth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 1475–1476, Toronto, Canada, 2010.

[59] Mingyu Guo and Vincent Conitzer. Optimal-in-expectation redistribution mechanisms. *Artificial Intelligence*, 174(5-6):363–381, 2010.

[60] Mingyu Guo and Vincent Conitzer. Strategy-proof allocation of multiple items between two agents without payments or priors. In *Proceedings of the Ninth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 881–888, Toronto, Canada, 2010.

[61] Mingyu Guo, Vincent Conitzer, and Daniel M. Reeves. Competitive repeated allocation without payments. In *Proceedings of the Fifth Workshop on Internet and Network Economics (WINE)*, Rome, Italy, 2009.

[62] Mohammad T. Hajiaghayi, Robert Kleinberg, and Tuomas Sandholm. Automated online mechanism design and prophet inequalities. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, Vancouver, BC, Canada, 2007.

[63] B. Paul Harrenstein, Mathijs M. de Weerdt, and Vincent Conitzer. A qualitative Vickrey auction. In *Proceedings of the ACM Conference on Electronic Commerce (EC)*, pages 197–206, Stanford, CA, USA, 2009.

[64] John William Hatfield and Paul R. Milgrom. Matching with contracts. *American Economic Review*, 95(4):913–935, September 2005.

[65] Bengt Holmström. Groves' scheme on restricted domains. *Econometrica*, 47(5):1137–1144, 1979.

[66] Hugo A. Hopenhayn and Christine Hauser. Trading favors: Optimal exchange and forgiveness. 2004 Meeting Papers 125, Society for Economic Dynamics, 2004.

[67] Leonid Hurwicz. On the existence of allocation systems whose manipulative Nash equilibria are Pareto optimal, 1975. Presented at the 3rd World Congress of the Econometric Society.

[68] Nathanaël Hyafil and Craig Boutilier. Partial revelation automated mechanism design. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pages 72–78, Vancouver, BC, Canada, 2007.

[69] Atsushi Iwasaki, Vincent Conitzer, Yoshifusa Omori, Yuko Sakurai, Taiki Todo, Mingyu Guo, and Makoto Yokoo. Worst-case efficiency ratio in false-name-proof combinatorial auction mechanisms. In *Proceedings of the Ninth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, Toronto, Canada, 2010.

[70] Matthew O. Jackson and Hugo F. Sonnenschein. The linking of collective decisions and efficiency. Working Papers 1159, California Institute of Technology, Division of the Humanities and Social Sciences, 2003.

[71] Radu Jurca and Boi Faltings. Minimum payments that reward honest reputation feedback. In *Proceedings of the ACM Conference on Electronic Commerce (EC)*, Ann Arbor, MI, USA, 2006.

[72] Ian A. Kash, Eric J. Friedman, and Joseph Y. Halpern. Optimizing scrip systems: efficiency, crashes, hoarders, and altruists. In *Proceedings of the ACM Conference on Electronic Commerce (EC)*, pages 305–315, San Diego, CA, USA, 2007.

[73] Ian A. Kash, Eric J. Friedman, and Joseph Y. Halpern. Manipulating scrip systems: Sybils and collusion. In *AMMA*, volume 14, pages 13–24, 2009.

[74] Sebastien Lahaie, David M. Pennock, Amin Saberi, and Rakesh V. Vohra. Sponsored search auctions. In Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay Vazirani, editors, *Algorithmic Game Theory*, chapter 28. Cambridge University Press, 2007.

[75] Ron Lavi, Ahuva Mu'alem, and Noam Nisan. Towards a characterization of truthful combinatorial auctions. In *Proceedings of the Annual Symposium on Foundations of Computer Science (FOCS)*, pages 574–583, 2003.

[76] Anton Likhodedov and Tuomas Sandholm. Mechanism for optimally trading off revenue and efficiency in multi-unit auctions. In *Proceedings of the ACM Conference on Electronic Commerce (EC)*, New York, NY, USA, 2004. Short paper. Earlier versions in AMEC-03 and ACMEC-03.

[77] Anton Likhodedov and Tuomas Sandholm. Methods for boosting revenue in combinatorial auctions. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pages 232–237, San Jose, CA, USA, 2004.

[78] Anton Likhodedov and Tuomas Sandholm. Approximating revenue-maximizing combinatorial auctions. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, Pittsburgh, PA, USA, 2005.

[79] Fangzhen Lin and Pingzhong Tang. Computer-aided proofs of Arrow's and other impossibility theorems. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pages 114–119, Chicago, IL, USA, 2008.

[80] Benjamin Lubin and David C. Parkes. Quantifying the strategyproofness of mechanisms via metrics on payoff distributions. In *Proceedings of the 25th Annual Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 349–358, Montreal, Canada, 2009.

[81] R. Preston McAfee. Amicable divorce: Dissolving a partnership with simple mechanisms. *Journal of Economic Theory*, 56(2):266–293, April 1992.

[82] Markus M. Möbius. Trading favors. May 2001. Manuscript.

[83] Dov Monderer and Moshe Tennenholtz. Optimal auctions revisited. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pages 32–37, Madison, WI, USA, July 1998.

[84] Hervé Moulin. Almost budget-balanced VCG mechanisms to assign multiple objects. *Journal of Economic Theory*, 144(1):96–119, 2009.

[85] Roger Myerson. Incentive compatibility and the bargaining problem. *Econometrica*, 41(1), 1979.

[86] Roger Myerson. Optimal auction design. *Mathematics of Operations Research*, 6:58–73, 1981.

[87] Roger Myerson and Mark Satterthwaite. Efficient mechanisms for bilateral trading. *Journal of Economic Theory*, 28:265–281, 1983.

[88] Abraham Othman and Tuomas Sandholm. How pervasive is the Myerson-Satterthwaite impossibility? In *Proceedings of the Twenty-First International Joint Conference on Artificial Intelligence (IJCAI)*, pages 233–238, Pasadena, CA, USA, 2009.

[89] Szilvia Papai. Strategyproof and nonbossy multiple assignments. *Journal of Public Economic Theory*, 3(3):257–71, 2001.

[90] David Parkes, Jayant Kalagnanam, and Marta Eso. Achieving budget-balance with Vickrey-based payment schemes in exchanges. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1161–1168, Seattle, WA, USA, 2001.

[91] David C. Parkes. When analysis fails: Heuristic mechanism design via self-correcting procedures. In *SOFSEM '09: Proceedings of the 35th Conference on Current Trends in Theory and Practice of Computer Science*, pages 62–66, 2009.

[92] David C. Parkes and Quang Duong. An ironing-based approach to adaptive online mechanism design in single-valued domains. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, Vancouver, BC, Canada, 2007.

[93] Steve Phelps, Peter McBurnley, Simon Parsons, and Elizabeth Sklar. Co-evolutionary auction mechanism design. *Lecture Notes in AI*, 2531, 2002.

[94] Ryan Porter, Yoav Shoham, and Moshe Tennenholtz. Fair imposition. *Journal of Economic Theory*, 118:209–228, 2004.

[95] Ariel D. Procaccia and Moshe Tennenholtz. Approximate mechanism design without money. In *Proceedings of the ACM Conference on Electronic Commerce (EC)*, pages 177–186, Stanford, CA, USA, 2009.

[96] Baharak Rastegari, Anne Condon, and Kevin Leyton-Brown. Revenue monotonicity in combinatorial auctions. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pages 122–127, Vancouver, BC, Canada, 2007.

[97] Tuomas Sandholm. Limitations of the Vickrey auction in computational multiagent systems. In *Proceedings of the Second International Conference on Multi-Agent Systems (ICMAS)*, pages 299–306, Keihanna Plaza, Kyoto, Japan, December 1996.

[98] Tuomas Sandholm, Vincent Conitzer, and Craig Boutilier. Automated design of multistage mechanisms. In *Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1500–1506, Hyderabad, India, 2007.

[99] Tuomas Sandholm and Andrew Gilpin. Sequences of take-it-or-leave-it offers: Near-optimal auctions without full valuation revelation. In *Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 1127–1134, Hakodate, Japan, 2006.

[100] Yuliy Sannikov. Games with imperfectly observable actions in continuous time. *Econometrica*, 75(5):1285–1329, 09 2007.

[101] James Schummer and Rakesh V. Vohra. Mechanism design without money. In Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay Vazirani, editors, *Algorithmic Game Theory*, chapter 10. Cambridge University Press, 2007.

[102] Taiki Todo, Atsushi Iwasaki, Makoto Yokoo, and Yuko Sakurai. Characterizing false-name-proof allocation rules in combinatorial auctions. In *Proceedings of the Eighth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 265–272, 2009.

[103] William Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *Journal of Finance*, 16:8–37, 1961.

[104] Yevgeniy Vorobeychik, Christopher Kiekintveld, and Michael Wellman. Empirical mechanism design: Methods, with application to a supply chain scenario. In *Proceedings of the ACM Conference on Electronic Commerce (EC)*, Ann Arbor, MI, USA, 2006.

[105] Yevgeniy Vorobeychik, Daniel Reeves, and Michael Wellman. Constrained automated mechanism design for infinite games of incomplete information. In *Proceedings of the 23rd Annual Conference on Uncertainty in Artificial Intelligence (UAI)*, Vancouver, BC, Canada, 2007.

[106] Liad Wagman and Vincent Conitzer. Optimal false-name-proof voting rules with costly voting. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pages 190–195, Chicago, IL, USA, 2008.

[107] Peter Wurman, Michael Wellman, and William Walsh. The Michigan Internet AuctionBot: A configurable auction server for human and software agents. In *Proceedings of the Second International Conference on Autonomous Agents (AGENTS)*, pages 301–308, Minneapolis/St. Paul, MN, USA, May 1998.

[108] Makoto Yokoo. The characterization of strategy/false-name proof combinatorial auction protocols: Price-oriented, rationing-free protocol. In *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 733–742, Acapulco, Mexico, 2003.

[109] Makoto Yokoo, Toshihiro Matsutani, and Atsushi Iwasaki. False-name-proof combinatorial auction protocol: Groves mechanism with submodular approximation. In *Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 1135–1142, Hakodate, Japan, 2006.

[110] Makoto Yokoo, Yuko Sakurai, and Shigeo Matsubara. Robust combinatorial auction protocol against false-name bids. *Artificial Intelligence*, 130(2):167–181, 2001.

[111] Makoto Yokoo, Yuko Sakurai, and Shigeo Matsubara. The effect of false-name bids in combinatorial auctions: New fraud in Internet auctions. *Games and Economic Behavior*, 46(1):174–188, 2004.

[112] Lin Zhou. Inefficiency of strategy-proof allocation mechanisms in pure exchange economies. Cowles Foundation Discussion Papers 954, Cowles Foundation, Yale University, September 1990.

Biography

Mingyu Guo was born in Wuhan, Hubei, China on September 10th, 1981. He received a B.S. in Mathematics from Zhejiang University in 2004, and a M.S. in Applied Mathematics from the University of Florida in 2006. From 2006 to 2010, he studied at Duke University for a Ph.D. in Computer Science. He was awarded the Outstanding Research Initiation Project Award from the Department of Computer Science, Duke University in 2008.

Mingyu Guo's research lies in the intersection of computer science and economics. His research interests include computational microeconomics, algorithmic game theory, multiagent systems, electronic commerce, auction theory, mechanism design, and prediction markets. During his Ph.D. studies, he coauthored more then ten peer-reviewed technical papers in his research area.